System Wide Information Management (SWIM)

Solution Guide for Segment 2A

Version: 1.0

25 June 2013

Submitted:	______________________________	______________
	SWIM System Engineering Lead	Date
Approved:	______________________________	______________
	SWIM Program Manager	Date

Federal Aviation Administration
800 Independence Avenue
Washington, D.C. 20591

DOCUMENT CHANGE HISTORY

Version	Date	Description of Changes
1.0	06/25/2013	Initial Document

Abstract

This System-Wide Information Management (SWIM) Solution Guide provides a high-level description of National Airspace System (NAS) Enterprise Messaging Services (NEMS) and a description of how NEMS services and features can be used by Federal Aviation Administration (FAA) programs that plan to provide service content or consume services.

This document was originally produced for the FAA by The Mitre Corporation, McLean VA, under Project No.: 0213BB05-AA, Outcome No.: 05, by the following team:

Duncan Thomson
Vikram Prabhu
Ezra Jalleta
Krishnan Balakrishnan

Acknowledgments

In producing this SWIM Solution Guide, The MITRE Corporation (MITRE) relied on existing material from the FAA, as well as detailed discussions with the NEMS service provider, the Harris Corporation. MITRE gratefully acknowledges the help, guidance, and the material provided by these organizations. In particular MITRE wishes to acknowledge Jim Robb of the FAA, Ed Zakrzewski of the ITT Exelis Corporation, Alex Prikhodko of North Star Group, and Steve Link, Tom Schabowsky, Gerry Batista, and Dave Almeida of the Harris Corporation.

Table of Contents

List of Figures

List of Tables

1 Introduction

System Wide Information Management (SWIM) enables the diverse legacy and new systems within the Next Generation Air Transportation System (NextGen) to deliver the right information to the right place at the right time. SWIM achieves this by providing an Information Technology (IT) infrastructure that enables systems to publish information of interest to National Airspace System (NAS) users, to request and receive information from other systems, and to do this in a way that supports Federal Aviation Administration (FAA) requirements for security and availability.

In Segment 2a, planned for implementation Fiscal Year (FY) 2012–2015, the centerpiece of the IT infrastructure being provided by the SWIM program is the NAS Enterprise Messaging Service (NEMS). NEMS provides messaging capabilities as a service commodity that can be provisioned to meet the needs of FAA programs in a similar fashion as other FAA telecommunications services. These messaging capabilities are integrated with existing NAS telecommunications, networking, and security infrastructure, in order to create a logical SWIM Enterprise Service Bus (ESB).

1.1 Purpose

The purpose of this document is to provide FAA program managers and architects with an understanding of how the SWIM ESB provided by NEMS can be used to meet program needs. The document is also intended to provide material that program managers can provide to contractors who will be bidding to build NAS systems that will need to use NEMS.

The information in this document is provided in the form of high-level descriptions of “solutions” that can be used by FAA programs to meet needs for information sharing that arise in different use cases.

1.2 Benefits

The solutions described in this guide offer advantages both to individual FAA programs that leverage these solutions, as well as to the FAA as a whole. From the perspective of a user program responsible for developing a system that needs to exchange information with other systems, the advantages of using NEMS include:

- Leveraging functionality that keeps track of which NAS and external systems need to receive which NAS information products, and routing those products to the necessary consumers. This relieves the user program from having to develop this functionality, and to maintain and update it as the NAS evolves.
- Leveraging a common solution for information system security needs, compliant with FAA security policy and approved by the NAS Information System Security Manager (ISSM).
- Leveraging reusable solutions for other recurring needs, such as solutions for failure detection and recovery.

From the perspective of the FAA as a whole, the advantages provided by having programs leverage SWIM solutions include:

- Decoupling the producers from the consumers, allowing, for example, new consumers to be added or deleted without impacting the producers. SWIM mediation services (described in Section 4) make it possible for one end system to change protocols or message formats without impacting other systems, making it easier to evolve the NAS incrementally.

- Limiting the variety of different protocols and message exchange patterns used for information dissemination within the enterprise. This will simplify the NAS system-of-systems architecture, facilitating the evolution of NAS to NextGen.
- Standardizing interfaces across the NAS will facilitate insertion of new systems into the NAS.
- Ensuring that NAS information products are known and potentially available for other possible uses within the enterprise.
- Making NAS information products available via a single logical point (the SWIM ESB), using a single process (the SWIM on-ramping process).
- Increased infrastructure status awareness as a result of NEMS ability to monitor all information exchange on the ESB.

1.3 Scope

The SWIM Solution Guide provides a high-level description of NEMS services and a general description of how those services can be used to meet a variety of different potential needs.

The Solution Guide focuses on the NEMS services that are being provided in SWIM Segment 2a. All of the NEMS services described in this document are available to be ordered by user programs. Some of these services have already been ordered by at least one program and are in operational use today; others are available to be ordered and can be put into operation within a short time of being ordered. All described services are expected to be available by the end of 2014.

1.4 Document Organization

Section 1 provides an introduction.

Section 2 provides information on the SWIM vision within NextGen and a brief overview of the SWIM program including its capabilities and infrastructure. It also introduces the processes that FAA programs will use to work with the SWIM program to order services to meet program needs.

Sections 3 describes some very general use cases that describe different kinds of information sharing needs that user programs are likely to have. For each use case, Section 3 provides a high-level description of a solution showing how these information-sharing needs can be met using the SWIM ESB.

For simplicity of exposition, the solutions described in Section 3 omit discussion of some important topics such as security and failure recovery. These topics are covered in Section 4.

Appendix A provides background information on some of the important underlying technologies being employed by SWIM: Java Messaging Service (JMS) and web services (WS). Appendix B provides a brief description of the SWIM service lifecycle management. Appendix C provides a detailed description of NEMS services. Appendix D provides a glossary and an acronym list.

1.5 How to Use this Solution Guide

Readers already familiar with the SWIM program may wish to skim the overview information in Section 2, paying special attention to the section that covers the terminology being used in this document (Section 2.2) and the section that describes the on-ramping process that is used by a program to obtain NEMS services (Section 2.7).

Readers less familiar with SWIM are advised to read all of Section 2 and perhaps familiarize themselves with the appendices.

Next, readers may wish to read the descriptions of use cases and solutions in Section 3, and identify those use cases and solutions that apply to their program. This will help the reader focus on the material in Section 4 that is applicable to their needs.

This Solution Guide is part of a planned series of documents. Design and programming guides are being developed in parallel with this document. Other planned documents will provide more design and ordering details to system architects and program managers, as well as programming information for software architects and developers planning to implement SWIM services in end systems.

In addition to these guides, readers are referred to the variety of technical and non-technical documents available on the *FAA SWIM Website* [1].

2 SWIM Overview and NEMS Capabilities

This section provides an overview of the SWIM vision and its benefits. It describes SWIM capabilities and the attendant infrastructure that provides these capabilities. It considers the SWIM service lifecycle management stages for potential SWIM business service producers and consumers. The concept of NEMS "on-ramping" along with specific features is provided. The use of these features is illustrated in Sections 3 and 4.

This section is based on material from the FAA SWIM program office, SWIM support contractors, and the FAA Telecommunications Infrastructure (FTI) service provider. The main sources used are the *SWIM and NEMS Familiarization Package* [2], *NEMS Ordering Concepts and Terminology* [3], *SWIM Service Lifecycle Management Processes v1.0 and v2.0* [4], [5], *NEMS Quarterly Program Review* [6], and the *FAA SWIM Website* [1].

2.1 SWIM Vision and Benefits

SWIM includes *standards, infrastructure,* and *governance* that enable net-centric Air Traffic Management (ATM) operations and information exchange between NAS entities and their stakeholders via interoperable services. Stakeholders include FAA mission support, service providers, airlines, other United States (U.S.) government agencies, and foreign Air Navigation Service Providers (ANSPs).

SWIM provides the basis for information exchange between systems based on the principles of a Service Oriented Architecture (SOA). SOA is a way of organizing IT assets, policies, practices, and frameworks that enable application functionality to be provided and consumed as services that can be invoked, published, and discovered. SOA makes an organization's IT better suited for interoperability among heterogeneous environments; one can interconnect between organizations regardless of their supported infrastructure, which opens doors to delegation, sharing, and reuse of existing services. This simplifies building interfaces to existing end systems and ensures new systems and applications can be created and more quickly integrated in order to create the new functionality needed for NextGen.

Further, SWIM provides governance to user programs to ensure services are SWIM compliant and meet all FAA SOA standards. By providing this governance and a supporting common enterprise infrastructure, SWIM reduces the cost and risk of rework for NextGen programs that develop and deploy services within the NAS.

A conceptual three-layered framework to describe the sharing of information via SWIM is illustrated in Figure 2-1. The conceptual layers are:

- **Application End Systems** that support the ATM mission. These systems support ATM operators (e.g., en route and terminal controllers, command center operators) and non-FAA airspace users and mission partners (e.g., airlines, other government agencies, other ANSPs). In order to implement NextGen, these application systems need to share information and access each other's services. With SWIM, they accomplish this by producing and consuming SWIM business services.
- **SWIM Infrastructure** provides the core services, including messaging infrastructure and governance, which allows application end systems to share information by producing and consuming each other's business services.

- **Network Infrastructure**, which consists of the NAS Operations Internet Protocol (IP) network and its component services.

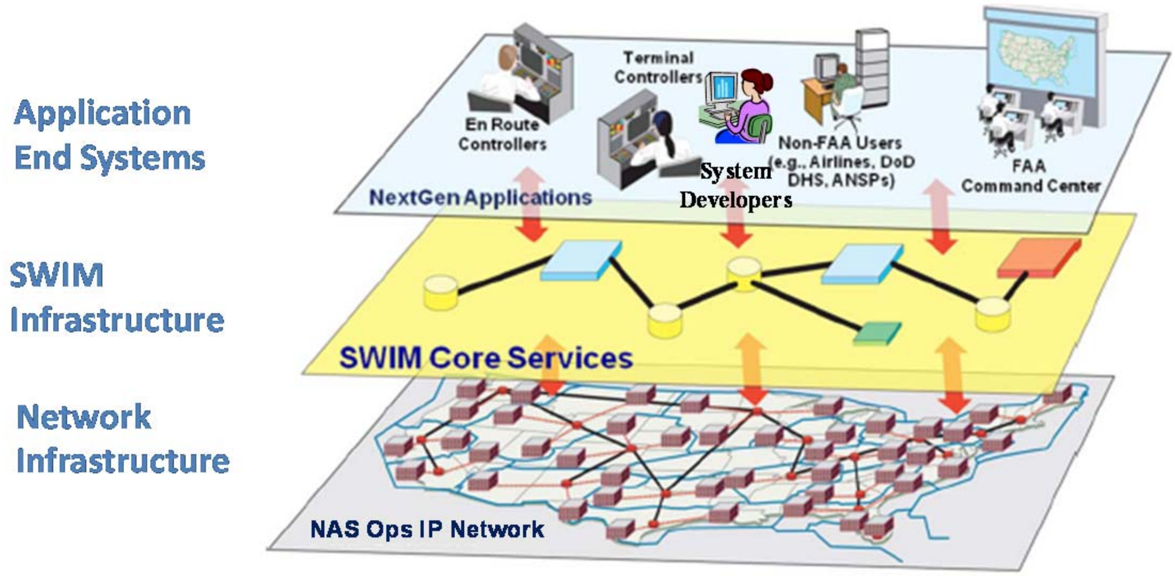


Figure 2-1. SWIM Conceptual Framework

2.2 SWIM Terminology

This section introduces some of the terminology used in this Solution Guide and in other SWIM documentation, in particular the *NEMS Ordering Concepts and Terminology* [3]. The terms in **bold font** are key terms that are used throughout this document. Only selected terms that are the most important to understanding the concepts in this document are included here, but a complete list of terms is provided in the Glossary. Readers unfamiliar with SWIM may find the explanation of terminology provided here somewhat terse; however, more detailed expositions will be found when the terms are introduced in the relevant sections later in the document.

The **NAS Enterprise Messaging Service (NEMS)** is a SWIM core service that provides messaging capabilities as a service commodity. NEMS is instantiated at a number of locations that we refer to as **NEMS nodes**. These provide the messaging infrastructure within SWIM, which we refer to as the **SWIM Enterprise Service Bus (ESB)**. In other SWIM documents, the reader may see the SWIM ESB referred to as the **NEMS ESB** or **NAS ESB—**these terms may have slightly different connotations, but all refer to the same logical information-sharing infrastructure.

The SWIM ESB is used to support the information sharing needs of the systems that provide the air traffic management, air traffic control, and supporting functions that make up NextGen. We refer to these systems as **application end systems**, or simply **end systems**.

To support the functions of NextGen, end systems will often need to make information available to other end systems. End systems accomplish this by making **SWIM business services** available that can be accessed by other end systems via the SWIM ESB. (Where the connotation is clear, the term **business service** may be used instead of SWIM business service.)

End systems that produce information or provide services are referred to as **producer systems**, or simply **producers**. End systems that consume information or invoke services are referred to as **consumer systems**, or simply **consumers**. To keep the explanations from becoming overly complex, the examples in this paper are written as if each end system is either a producer or a consumer, but the reader should bear in mind that in reality an end system may be both a producer as well as a consumer. (End systems may also support other system interactions that are not SWIM enabled.)

We refer to the organizational units that are responsible for application end systems as **user programs**. User programs will need to interact with the Communications Information and Network Programs (CINP) group (AJM-31) to plan, integrate, and operate end systems that will use the SWIM ESB. A key interaction between a user program and CINP is the **on-ramping process**. This is a two-phase process in which candidate SWIM business services are first **qualified** (including integration testing and verification with SWIM infrastructure) and then ordered and provisioned to support the end systems. During the qualification phase, business services are planned and registered in the NAS Service Registry/Repository (NSRR); each SWIM business service is then qualified for operations on the NEMS infrastructure. In the second, operational phase of the on-ramping process for a producer system, one or more **producer on-ramping services (POSs)** will be ordered to allow the system to make its business services available to other systems via the SWIM ESB. Similarly, during the operational on-ramping phase for a consumer system, one or more **Consumer On-Ramping Services (COSs)** will be ordered to allow the system to consume business services via the SWIM ESB. In addition, an **Exchange Service** will need to be provisioned for each SWIM business service to provide configuration management and reporting of the brokering of information exchanges between producers and consumers. While there is only one service qualification and only one Exchange Service per SWIM business service, many producer instances may contribute to a particular business service and each producer instance requires its own POS.

Note that, while we are primarily concerned with NAS end systems, the SWIM ESB also supports information sharing between NAS and non-NAS[1] systems (e.g., airline systems), so in this document an end system may be either a NAS or a non-NAS system. The **NAS Boundary Protection Service (NBPS)** is an enterprise security service that allows NAS systems to interoperate with non-NAS systems by allowing approved information exchanges through any of four NEMS enabled **NAS Enterprise Security Gateways (NESGs)** that are the egress points for the NBPS.

2.3 SWIM Program Overview

In SWIM Segment 1, the SWIM program used a federated approach and did not implement any enterprise-wide messaging infrastructure. Each SWIM Implementing Program (SIP) was thus responsible for its own message delivery. SWIM specified the standards, set up the NSRR[2], and provided funding along with a standard software package to SIPs to implement messaging within their systems. This was the service container software from FuseSource.[3]

In Segment 2a, planned for implementation Fiscal Year (FY) 12-15, SWIM is deploying an ESB that is composed of NEMS nodes as well as other supporting infrastructure components. The SWIM ESB is integrated with existing NAS telecommunications and IT infrastructure and

[1] In the context of this document, the term “non-NAS” refers to systems that have been designated as external for the purposes of boundary protection, as discussed in Section 4. It does not imply that these systems are not important to the overall functioning of the NAS.

[2] In SOA, a service registry is the means for producers to register their services, their description, and the interface standards for accessing the service. Similarly consumers will query the registry to discover what services are available, and their quality of service.

[3] FuseSource is currently a part of the Red Hat Corporation.

supplemented with agile governance. Systems built using the SWIM Segment 1 are now being transitioned to use the NEMS messaging infrastructure.

In SWIM Segment 2a, services such as messaging provided by the SWIM ESB are structured as commodity IT services that can be obtained using service provisioning processes similar to those established for the NAS Operations (Ops) IP network and dedicated circuit services. The NEMS infrastructure leverages FAA investments to provide a fully integrated service set to user programs. These include the NAS Ops IP Network, NBPS, and the NAS Enterprise Management System including the FAA Network Enterprise Management Centers (NEMCs), the FTI Network Operations Control Center (NOCC) and the FTI Security Operations Center (SOC). Based on their concept of operations, architecture and requirements, application end systems can implement system interfaces using NAS Ops IP, SWIM, or a combination of these, as illustrated in Figure 2-2.

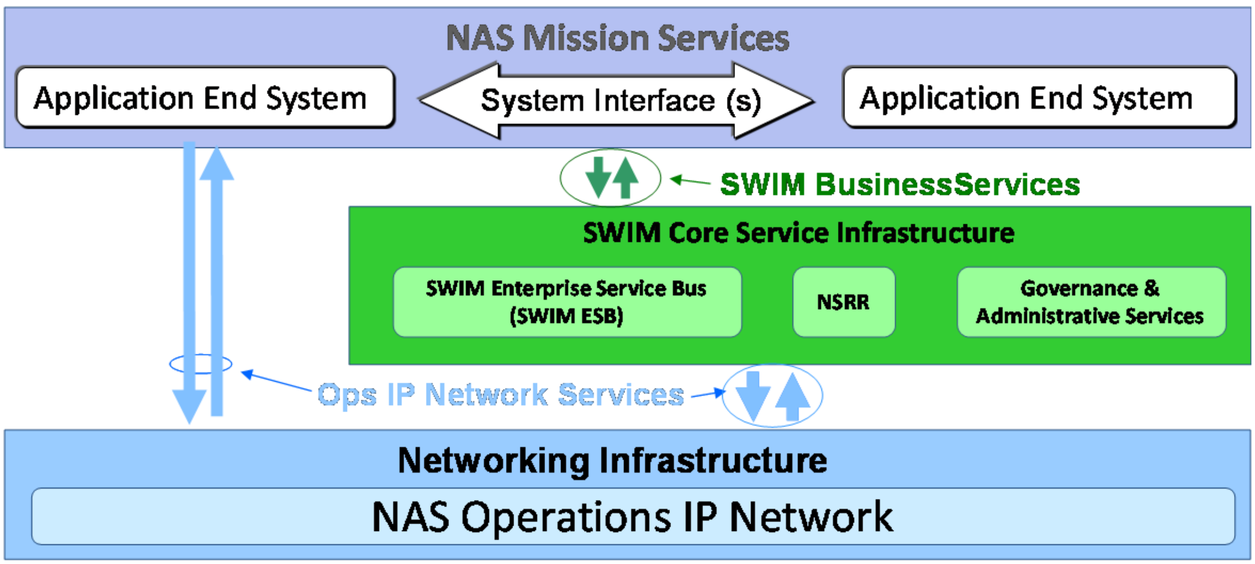


Figure 2-2. SWIM/NEMS Role in the NAS

2.4 SWIM 2a Capabilities

The SWIM Segment 2a capabilities include:

- **Enterprise Messaging Services** that provide a standard policy driven method of exchanging messages between a variety of NAS and non-NAS user programs. Both Java Message Service (JMS) & Web Service (WS) messaging protocols are supported. A brief introduction to JMS and WS is provided in Appendix A.
- **Information Assurance** that is provided by the NEMS by allowing granular control over which service providers and service consumers connect to the messaging infrastructure, and by enforcing access control to service capabilities. The NBPS has been enhanced with SWIM/SOA security controls and associated capabilities to support communication between the SWIM ESB and non-NAS systems.
- **SWIM Governance** that consists of a set of activities used in exercising control over business services across lifecycle stages. SWIM governance formulates appropriate policies, processes, and standards to achieve better alignment with FAA enterprise objectives through the use of flexible and modular SOA-based services. SWIM

governance allows for greater agility, and lowers the total cost of ownership through the creation, deployment, and reuse of shared services. The Service Lifecycle Management Process (SLMP) is an overarching process for enforcement of SWIM governance principles from conception to retirement of a SOA Service. User programs using the SWIM messaging infrastructure must follow the SLMP. SWIM governance and associated documentation of services in the NSRR provide the NAS-EA with tools for evaluating the NAS service assignments and authority allocated to various NAS programs.

2.5 SWIM Infrastructure in Segment 2a

SWIM Segment 1 implementation activities used a federated approach and the SWIM implemented infrastructure focused on the development of governance capabilities including a NAS-wide registry. In addition, two of the four NESGs of the NBPS were enhanced to provide NAS boundary protection for non-NAS SWIM consumers as well as IP communications. In general, SWIM Segment 2a continues to utilize these assets and adds a messaging infrastructure by leveraging the FTI service infrastructure.

2.5.1 NAS Service Registry/Repository (NSRR)

The NSRR is a NAS-wide registry that supports tracking and discovery of mission services. It is an authoritative source of information regarding services available through SWIM. A service enrollment in the NSRR involves a description of the data products available by subscription to the service, indication of the service lifecycle stage, and a host of supporting documents. Producers must enroll their services in the NSRR to allow for discoverability of these services by consumers. The NSRR can be used to discover available services as well as services in development. The NSRR is also used to catalog the web services that are provided and are available via request/response messaging.

2.5.2 NESG & NAS Boundary Protection

NESGs are key elements of NBPS that provide security controls between the NAS and non-NAS network communications. The NESGs at ACY and ATL were enhanced to support external consumers in Calendar Year (CY) 2011.[4] They are being further upgraded to support both external consumers and external producers in CY13; the NESGs in SLC and OEX are also being upgraded to support both external producers and consumers in CY13.

2.5.3 SWIM ESB

The messaging services are provided via the implementation of NEMS nodes at NAS Air Route Traffic Control Centers (ARTCCs) and at some other NAS facilities; the collection of these nodes, interconnected by the Ops IP network, constitutes the SWIM ESB. The NEMS nodes support COTS ESB products from Oracle and FuseSource.

The current deployment of NEMS includes NAS messaging nodes at ZTL, ACY, ZLC, and OEX facilities. Additional NEMS nodes will be configured and brought into service starting in FY13 with approximately four nodes added per year, the next four being at ZSE, ZFW, ZDC, and ZAU[5]. In addition, there are nodes in the Atlanta and Atlantic City NESGs that support NBPS;

[4] ATL is the Atlanta NEMC facility; ACY and OEX are the William J Hughes Technical Center at Atlantic City and the Mike Monroney Aeronautical Center at Oklahoma City. ZTL is the Atlanta ARTCC. ZLC is the Salt Lake City ARTCC.

[5] ZSE, ZFW, ZDC and ZAU are respectively the Seattle, Fort Worth, Washington, and Chicago ARTCCs.

the OEX and SLC NESGs are scheduled for upgrades with NEMS node compatibility. There are also NEMS nodes in the FTI National Test Bed (FNTB) and the Research and Development (R&D) domain that support prototyping and testing.

Specific NEMS features relevant to programs are described later in this section. Description of how programs can utilize these NEMS features is provided in Sections 3 and 4.

2.5.4 Network Connectivity

All connections discussed throughout this paper are made with Internet Protocol (IP) and the other SWIM standard protocols, which are described in *SWIM Segment 2 Technical Overview* [7]. For the most part, the details of IP network connectivity are not discussed in this solution guide, except that we do need to bear in mind that for security reasons NAS internal systems and networks are isolated from external systems and networks; this is addressed in more detail in Section 4. For NAS internal application end systems, connections to NEMS are made via the NAS Ops IP network; in some cases, these may be Local Area Network (LAN) services and in other cases they may be Wide Area Network (WAN) services—but this is transparent to the end systems. External application end systems will connect to NEMS via external connections to an NESG component of the NBPS. These external connections to a NESG may be implemented via encrypted dedicated circuits or encrypted virtual private network (VPN) connections over external networks provided by the user. Refer to the FTI Operational IP Users Guide [8] for more information on IP connectivity.

2.5.5 Domain Name System (DNS)

The DNS provides naming and name-to-address resolution services across the NAS Ops IP network. The SWIM ESB will leverage NAS infrastructure capabilities such as DNS (and Load Balancing in conjunction with DNS) to provide a more robust NEMS service, i.e., to provide producers and consumers a flexible and robust connectivity to the ESB. Producers and consumers will know they are connected to the SWIM ESB, but they are not required to know which specific NEMS node they are connected to at any specific instance of time.

2.5.6 Network Time Protocol/Precision Time Protocol (NTP/PTP)

The NTP/PTP provides time synchronization services across the NAS Ops IP network. NTP/PTP will be leveraged as additional value added services are incorporated.

2.5.7 R&D Domain and FNTB

The FAA maintains an R&D domain and the FNTB at the William J Hughes Technical Center (WJHTC). The R&D domain provides a protected prototyping environment for development and experimentation activities for FAA & NextGen sponsored R&D activities; it also provides access to NAS data available via SWIM services and limited sets of legacy data.

The R&D domain contains an instance of an NEMS-enabled NESG gateway, also known as the R&D Security Gateway, and an internal NAS NEMS node that supports virtualization to support multiple concurrent, but separate, research activities. Prototype producers and consumers may conduct experimentation and testing activities in the R&D domain, and can utilize the R&D security gateway and the R&D ESB for testing purposes to make sure that their systems operate as intended.

Development activities occur in the R&D Domain with transition to the FNTB for inter-operability testing activities prior to final deployment to NAS.

2.5.8 NAS Enterprise Management System

NAS Enterprise Management System includes the FAA NEMCs, the FTI NOCC, and the FTI SOC. The FTI operations centers are used to manage the FTI infrastructure and particularly the NAS Ops IP network. The use of these operations centers is being leveraged to manage the SWIM ESB capabilities as well. Leveraging of this capability allows the user programs to follow familiar FTI processes. It also allows for the integration of some NEMS monitor and control capabilities with that of FTI, the NAS Ops IP network, and the NBPS for troubleshooting and reporting. (Service monitoring is discussed in Section 4.)

2.6 SWIM Service Lifecycle Management (SLM)

Business services that are made available to the NAS via SWIM are enterprise services and are thus subject to strict governance rules that are applicable throughout their lifecycle. A brief summary of the SLM process is provided in Appendix B for an application program that is unfamiliar with SLM.

Proposals for business services may come from individual programs, Communities of Interest (COI), the NAS Enterprise Architecture (EA), and aviation user groups such as RTCA. In the NAS EA, these ideas are usually in support of the NextGen Operational Improvements that are being considered. During SWIM Segment 1, potential services were proposed and decided by ATM domain-centric COIs, i.e., Flow-flight, Weather, and Aeronautical.

Standards for information exchange for business services are subject-specific standards; these were developed by the FAA in conjunction with other ANSPs and international stakeholders in the aeronautical community. Examples of these information exchange standards are Flight Information Exchange Model (FIXM), Weather Information Exchange Model (WXXM), and Aeronautical Information Exchange Model (AIXM). Information on these standards and their usage is available at the *AIXM website* [9]; the AIXM website provides links to the FIXM and WXXM websites. Potential producers and consumers will have to conform to these standards or a combination of these standards.

The SLM process defines 7 stages for a service producer:[6]

- Proposal
- Definition
- Development
- Verification
- Production
- Deprecation
- Retirement

Service approval is the initial authorization to provide a SWIM-compliant service and subsequent approval of service artifacts and lifecycle advancement (e.g., from a development stage to a production stage, etc.). Service approval is contingent upon SWIM compliance. The SLM process describes the activities that should occur to confirm compliance with SWIM policies within each stage of the SLM process.

[6] A service consumer gets involved only in the later stages after a business service has been advertised and has reached final development and testing. Please see Appendix B for details.

SWIM and FAA policies require that all SWIM-compliant services be registered in the NSRR and that the meta-information describing the service be provided. Registration of service meta-information (meta-data and service artifacts) occurs throughout the SLM process, and must be submitted to the NSRR at the required lifecycle stage. Registration of a service and submission of associated meta-information within the NSRR promotes early discovery and information sharing. For the purposes of this document, the SWIM SLM should be considered a service configuration management process that has been integrated with the NEMS on-ramping process described in the following paragraphs.

Comprehensive treatment is available in SWIM Service Lifecycle Management Processes, v1.0 and v2.0 [4], [5].

2.7 NEMS On-Ramping

Once an end system program has received FAA service approval to proceed to service definition, questions on how to use the NEMS infrastructure and capabilities arise. These questions are addressed during the SWIM on-ramping process, in which the user program works with the SWIM program and the NEMS service provider to plan and provision the SWIM infrastructure services that will be used by the end system. A brief overview of this process is described in this section.

On-ramping creates an authorized association of a producer or consumer end system to the SWIM ESB in order to provide or consume business services. SWIM offers several basic types of on-ramping services for end systems, in which one or more service classes are ordered based on the following:

- Type of service—either JMS or WS.
- Reliability, Maintainability, Availability (RMA) requirements—either RMA3 or RMA4.[7]
- Security domain of end system—either internal or external to the NAS.

Additionally, enhanced features can be ordered as needed by the user program. These are described later and their use is illustrated in Sections 3 and 4.

Most of the NEMS nodes will be located at FAA ARTCCs. In early years, all ARTCCs may not have NEMS nodes. All end systems will connect to NEMS via the Ops IP network, regardless of a NEMS node being collocated or not. The SWIM program office will look at the aggregate of all producer and consumer locations and design an efficient rollout strategy to other ARTCCs, as needed.

2.7.1 Distribution of Service Content from Producer to Consumer

Figure 2-3 shows the end-to-end view of the distribution of products (business services) from a producer system to a consumer system. The POS represents the NEMS services ordered for the producer systems during the on-ramping process for the purposes of connecting to the SWIM ESB. The producer system uses the POS to make its information products and services available on the SWIM ESB. POS interactions between the producer and the SWIM ESB are implemented by communicating with a NEMS node via the Ops IP network. Also during the on-ramping process, an “Exchange Service” is ordered to provide the distribution of end system products

[7] RMA3 and RMA4 are different RMA service levels as defined in the *FTI Telecommunications Service Guide* [14].

from producers to consumers. This dissemination of products from producer to consumer is performed by the NEMS nodes that make up the SWIM ESB.

Similarly, the COS represents the NEMS connection services ordered for the consumer systems during the on-ramping process. The consumer's COS is (in a similar fashion to the producer's POS) is implemented by communicating with the SWIM ESB over the Ops IP network. The consumer uses the COS to receive the information products and services that it has subscribed to during the on-ramping process.

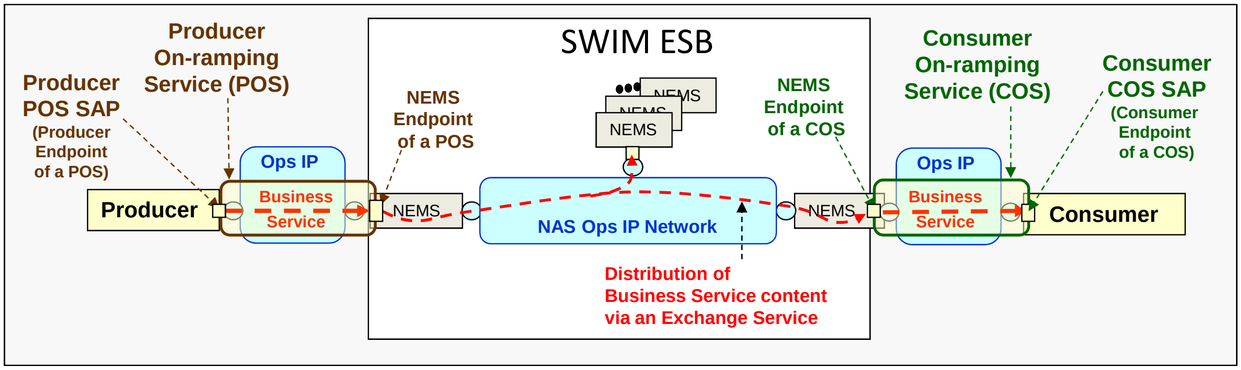


Figure 2-3. Distribution of Products from Producer to Consumer via an Exchange Service

SWIM will provision the NAS Ops IP services as necessary for its infrastructure to account for the access interactions with producers/consumers and for content transfers between various NEMS nodes in the SWIM ESB.

Exchange services can be used to associate specific subsets of business service content between producers and consumers. As an example (see Figure 2-4), consider two producers A and B that provide service content in the form of Product X1 and Product X2 (of the same business service X) respectively to the SWIM ESB. Both Product X1 and Product X2 contain subsets of information that consumers can subscribe to from the SWIM ESB. Consumer 1 is interested in all of product X1, Consumer 2 is interested in both products X1 and X2, and Consumer 3 wants only sub-product "c" of product X2. These subscriptions are maintained in the SWIM ESB, and the Exchange Service provides for and monitors the delivery of products from producers A and B to consumers 1, 2, and 3 accordingly.

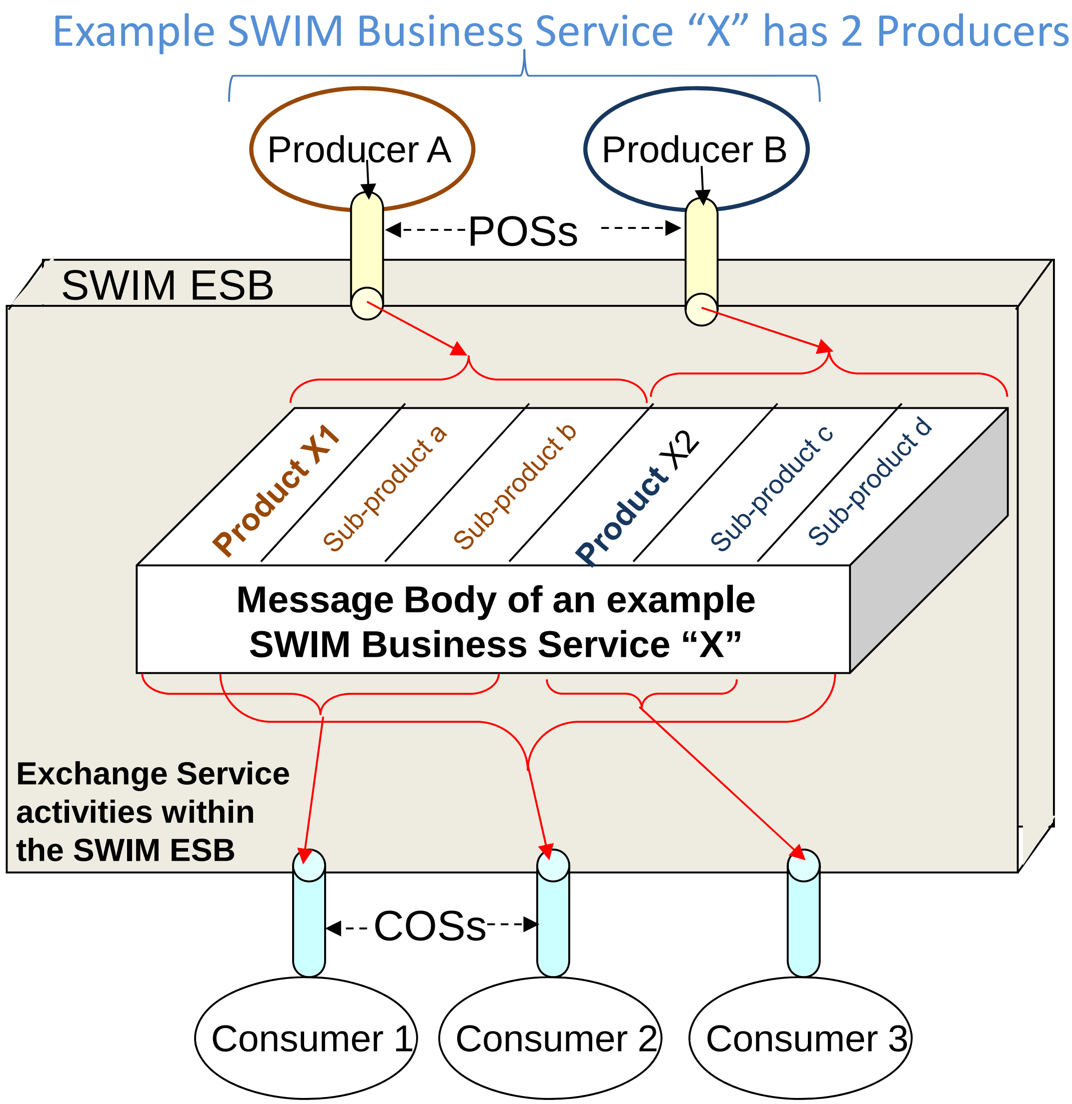


Figure 2-4. Exchange Service Associations

2.7.2 Sample Ordering Steps in On-Ramping

Details on NEMS service ordering are beyond the scope of this document. The general steps in on-ramping a producer and its products are as follows:

1. The producer orders the POS service with the appropriate options. One service is ordered for each producer-to-NEMS association that is required. A POS Unique Service Identifier (USI) is assigned to track/monitor producer activity (active, in tolerance, etc.).
2. Exchange services are ordered. One Exchange Service is ordered for each SWIM business service on-ramped. A business service can be on-ramped by one or more producers, but in all cases, only one Exchange Service is required, even if multiple producers contribute to the business service. An Exchange Service USI is assigned to control and monitor distribution of content to consumers. The Exchange Service USI associates the Exchange Service to a particular business service as defined in the NSRR.

3. Consumer programs can identify a new SWIM business service in the NSRR and either add a new consumer to access the service content or add content of the new service to an existing consumer COS.

An Exchange Service relates producers to consumers. When initially created via the producer on-ramping process, only producer relationships exist on the Exchange Service. Consumer relationships are added to the Exchange Service as one or more consumers on-ramp using the COS and request content of that particular SWIM business service.

Producer and consumer programs should provision sufficient NAS Operations IP bandwidth as required for transport of data flows to/from the SWIM ESB.

2.8 NEMS On-Ramping Services/Features

On-ramping service options that can be ordered by producers and consumers are categorized into basic and enhanced services. A brief description of these on-ramping services is provided in Table 2-1. The terms U1 through U4 and E1 through E9 are NEMS designations to identify basic and enhanced on-ramping services. Appendix C provides a comprehensive description of all NEMS on-ramping capabilities.

2.9 Service Provisioning

NEMS service provisioning procedures are streamlined because NEMS on-ramping services are part of the FTI contract and are ordered and tracked using the same tools as regular telecommunication services. Details of ordering are beyond the scope of this document.

Procurement of NEMS on-ramping services and telecommunication services is overseen by the FAA's Enterprise Engineering Services organization, which will utilize SWIM and FTI engineering staff (FAA and contractor) to provide needed expertise. The responsibilities of these groups are:

- The Enterprise Engineering Services organization oversees the engagement with user programs, and ensures that system architectures and design objectives meet enterprise goals.
- SWIM engineering staff design the network topology, node sizing and rollout strategy, manage on-ramping of producers and consumers, ensure NEMS performance, etc.
- SWIM engineering staff provide SOA-related expertise, enforce SWIM governance requirements, ensure NSRR enrollment, and provide cost estimation for SOA business services.
- FTI engineering staff provide telco-related expertise, bandwidth estimation, network architecture design, and cost estimation for telco services.

Table 2-1. NEMS On-Ramping Services/Features

Service or Feature Type	NEMS Designation	Name of Service/Feature	Description
Basic	U1	WS-C Web Service-Consumer	On-ramping of a consumer in order for the consumer to access web services being proxied by the NEMS
	U2	WS-P Web Service-Producer	On-ramping of a producer in order to make web services available to authorized NEMS consumers
	U3	JMS-C JMS Subscription-Consumer	On-ramping of a consumer in order for the consumer to access JMS subscription services being offered by the NEMS
	U4	JMS-P JMS Publishing-Producer	On-ramping of a producer for publishing content to the NEMS via JMS
Enhanced	E1	Mediation	Message content and protocol transformation capabilities for NEMS producers and consumers
	E2	Performance	Guaranteed message delivery for JMS and WS consumers as well as durable subscriptions for JMS if required
	E3	Service Orchestration	Development and deployment of Business Process Execution Language (BPEL) driven service orchestration flows utilizing currently deployed services
	E4	SLAs	Monitoring, reporting and management services to provide Service Level Agreement (SLA) monitoring
	E5	Security	Advanced service level access control including integrity, privacy, and encryption for NEMS services specifically WS and JMS
	E6	Interoperability	Provides capability for the NEMS to interoperate with a producer or consumer via a Java Connector Architecture (JCA) adaptor
	E7	Reports	Capability to generate custom reports providing summary and detailed performance metrics beyond the information provided by standard reports
	E8	Dynamic Subscriptions	Supports a subscription service managed by a producer which enables initiation of dynamic subscriptions by a consumer for consumer specified content
	E9	Publications to R&D	Publishing data from operational network to R&D domain for non-operational consumer connection and consumption

2.10 Service Costing

The SWIM Investment Planning and Analysis Team will work with potential SWIM users to develop cost estimates for on-ramping services onto the NEMS. Cost estimates are based on:

- Identification of required basic and enhanced on-ramping services that are ordered via FTI Contract Line Item Numbers (CLINs).
- Identification of the complexity factor, which will drive the number or percentage of CLINs exercised.
- Past experience from SWIM Segment 1 to help user programs identify service development costs (e.g., software lines of code for clients, governance requirements) as well as costs to acquire software, training, and consulting services for SWIM compliant SOA software products).

The cost responsibilities are generally allocated as follows:

SWIM User	SWIM Program
• FTI services and bandwidth to access NEMS. • Basic and enhanced on-ramping services. • Business service development and sustainment costs. • Response to SWIM/NAS governance requirements. • Costs for hardware and software to publish services onto the NEMS.	• NEMS messaging infrastructure. • Coordination with FTI to identify basic and enhanced on-ramping Services. • Intra-NEMS data transport. • Operation and support of NSRR. • SWIM governance.

2.11 Communication and Collaboration Activities

For existing and potential user programs that either use SWIM or wish to consider SWIM there are several venues for education, communication, and collaboration. These include:

2.11.1 SWIM Working Groups

The main SWIM working groups consist of the Architecture Working Group (AWG), the COTS Working Group (COTSWG), and the Service Registry Working Group.

- **SWIM Architecture Working Group.** The AWG is a technical forum with a purpose of sharing ideas about SOA implementation (service design and architecture). Its purpose is to harmonize architectures between SWIM and user programs, identify architectural issues and find solutions, and share lessons learned. AWG members include existing and future programs.
- **COTS Working Group.** The SWIM COTSWG is a forum for reviewing status of and planning SWIM COTS and Open Source product management, including risk management. It also establishes SWIM product/tool use policies and manages product/tool issues. COTSWG members include SWIM & program subject matter experts and vendor representatives (as needed). SWIM COTS information is available on the *SWIM Wiki* [10].

- **Service Registry Working Group.** The Service Registry Working Group works with user programs and SWIM stakeholders in identifying, classifying, and registering services.

2.11.2 Educational Activities

Educational activities include the regular SWIM SOA Brown Bag briefings on various topics; information on past topics is available on the *SWIM KSN* [11]. White papers and technical overview documents are also available on the *SWIM KSN*. SWIM also holds a “SWIM’posium” conference yearly for FAA and its stakeholders.

2.11.3 Collaboration Activities

Electronic collaboration, education, and information exchange activities are supported by the *SWIM Website* [1], the *SWIM Wiki* [10], and the *SWIM KSN* [11].

3 Basic SWIM Solutions for NAS Information Exchange Use Cases

The SWIM capabilities provided by NEMS can be used in different ways to support information exchange needs within the NAS and between the NAS and non-NAS systems. While many variations are possible, they fall into a relatively small set of solution patterns, each suitable for a different type of information exchange need. This section describes these basic solution patterns.

To keep the explanation simple, many important practical issues (e.g., failure handling, security, and monitoring) and special features (e.g., mediation and message quality of service) are not described here, but are covered in Section 4.

The solutions described in this section are presented from the point of view of a user program that will be using SWIM for information exchange. By considering different types of user program information exchange needs, we have arrived at a small set of use cases. These use cases are based on current or planned NEMS services and reflect needs or scenarios that potential users of NEMS services have expressed.

The use cases are:

1. **Publish/Subscribe**. Producer systems have information that must be pushed to one or more consumer systems whenever the information is updated or new information is available.
2. **Request/Response**. Consumer systems make dynamic requests to receive or update specific information products, or request specific processing services.
3. **Dynamic Subscription**. Consumer systems make dynamic requests for specific information products, and producer systems then push those products to the consumers whenever the information is updated or new information is available. (This is a “hybrid” of the two previous use cases.)

In the following sections, we describe each use case and show how that use case can be supported with solutions using SWIM services that are orderable within SWIM Segment 2a. These solutions leverage the SWIM ESB that is composed of NEMS nodes, as well as other enterprise services, including the Ops IP Network and security controls of the NBPS.

3.1 Publish/Subscribe Use Case

3.1.1 Use Case Overview

This use case is illustrated in Figure 3-1.

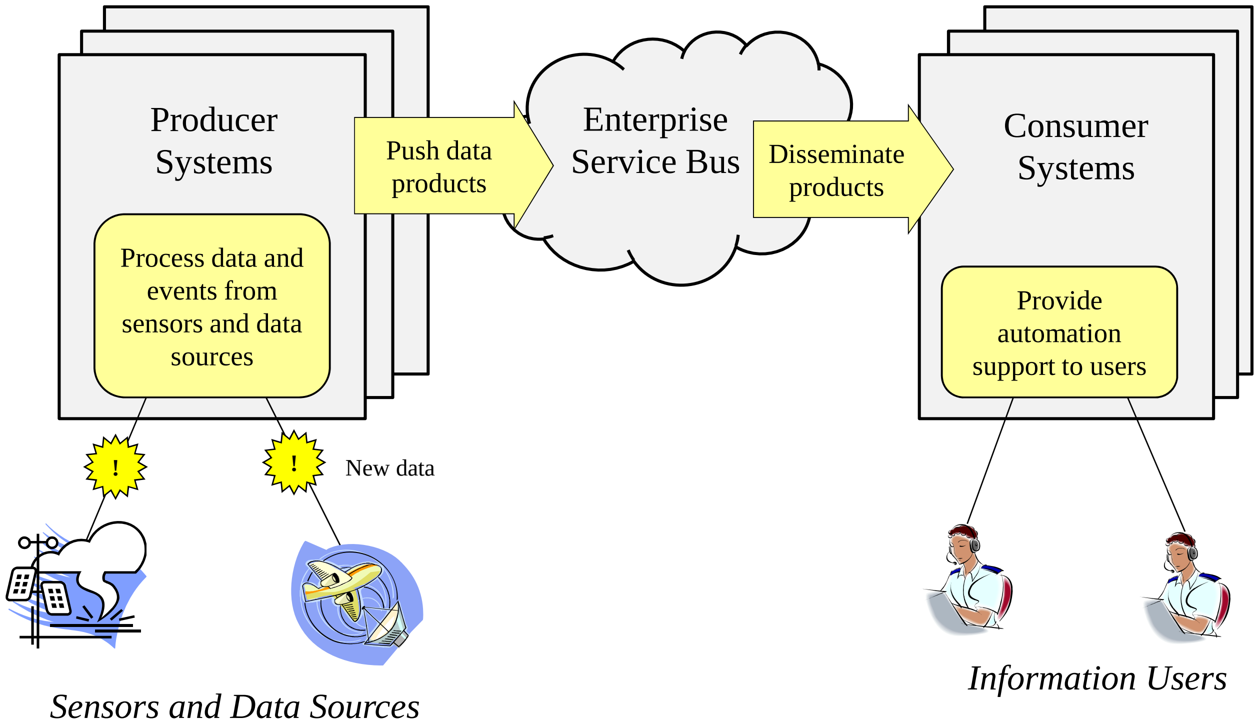


Figure 3-1. Publish/Subscribe Use Case Overview

In this use case, a user program needs to provide an ongoing stream of data messages from one or more producer systems to one or more consumer systems. As new or updated data is available, or whenever events occur, the producer systems will create messages that contain data products and event notifications. These messages will then be "pushed" out by the producer systems to the ESB for delivery to consumers.

An example of this use case is the surface movement event SWIM business service, which provides the airlines with business value by giving airline personnel real-time insight into operations happening on the airport surface. In this example, aircraft moving on the airport surface are tracked with Airport Surface Detection Equipment-Model X (ASDE-X) sensors, resulting in a stream of surface movement event data messages. These messages are published as a SWIM business service by the SWIM Terminal Data Distribution System (STDDS) in the Terminal Radar Approach Control facilities (TRACONs). The data is disseminated over the SWIM ESB and provided to multiple different airline systems that provide automation support to airline users.

Note that, in this use case, individual human users are not identified as consumers; the producer system provides the data to consumer systems, which are responsible for providing decision support and other automation functions needed by human users (providing the Graphical User Interface (GUI), identification, authentication and access control, etc.).

3.1.2 Publish/Subscribe Solution

3.1.2.1 Solution Overview

An overview of a solution for this use case is shown in Figure 3-2.

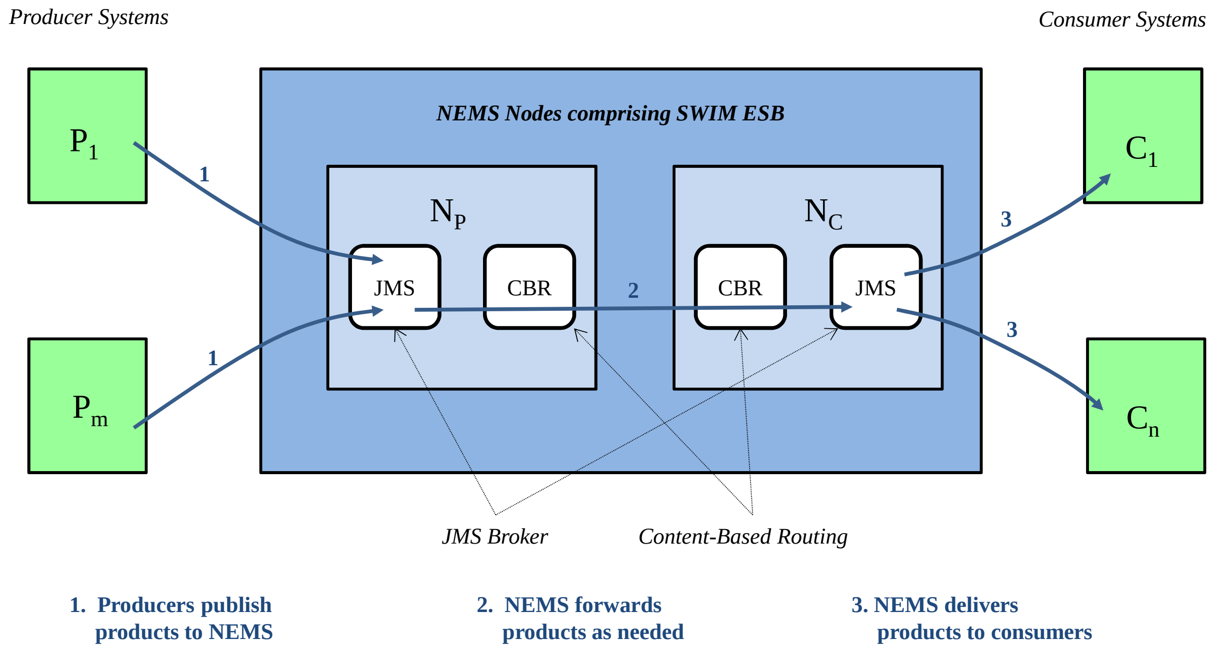


Figure 3-2. Publish/Subscribe Solution Overview

Figure 3-2 illustrates multiple producer systems (P_1 through P_m) that are producing information that is needed by a set of consumer systems (C_1 through C_n). To achieve this, each producer and consumer system communicates with the SWIM ESB by using IP network services to establish a logical JMS connection with a NEMS node. The producer systems provide a SWIM business service by packaging information into Extensible Markup Language (XML) formatted messages[8] and publishing these messages to a JMS queue on the SWIM ESB. The Content Based Routing (CBR) logic performed by the NEMS nodes is responsible for disseminating these XML messages through the SWIM ESB, based on the subscriptions that have been defined for each consumer, as described below in the section on product dissemination. Finally, the consumer systems access the SWIM business service by reading the messages from a JMS topic on the SWIM ESB. Each Consumer has its own dedicated topic.

Note that the CBR function is performed by the SWIM ESB. During operation, the CBR function is transparent to producer and consumer systems. The producers need only publish new information to the ESB, and the consumers need only retrieve information from the ESB.

In Figure 3-2, the producer systems are shown as communicating with NEMS node N_P, and the consumer systems are shown as communicating with NEMS node N_C. However, we should emphasize that this is just one possible configuration chosen for purposes of illustration. It may be that different producers/consumers onramp to different N_P/N_C, or that N_P or N_C are the same NEMS node, or it may be that there are additional intermediate NEMS nodes in between N_P and N_C. This is transparent to the producer and consumer systems—they need only connect to the SWIM ESB using the JMS queue or topic addressing information provided by SWIM during the on-ramping process.

[8] XML is the SWIM standard. Note that SWIM supports compression (by default) to mitigate any bandwidth concerns related to the use of XML. Readers interested in XML bandwidth issues are referred to Section 5 of *System Wide Information Management (SWIM) Evolution Considerations* [24].

3.1.2.2 Product Dissemination

During the producer program on-ramping process, the producer systems and the products that each producer system can publish are defined. During the consumer program on-ramping process, the consumer systems are identified and each consumer system subscription is configured to include the subset of the available products, as needed for that consumer system.

As part of the producer on-ramping process, producer programs need to provide a taxonomy for their products. This taxonomy will define the fields that can be used by the CBR process within the ESB to route messages to the appropriate set of consumers. Figure 3-3 illustrates the concept of product taxonomy.

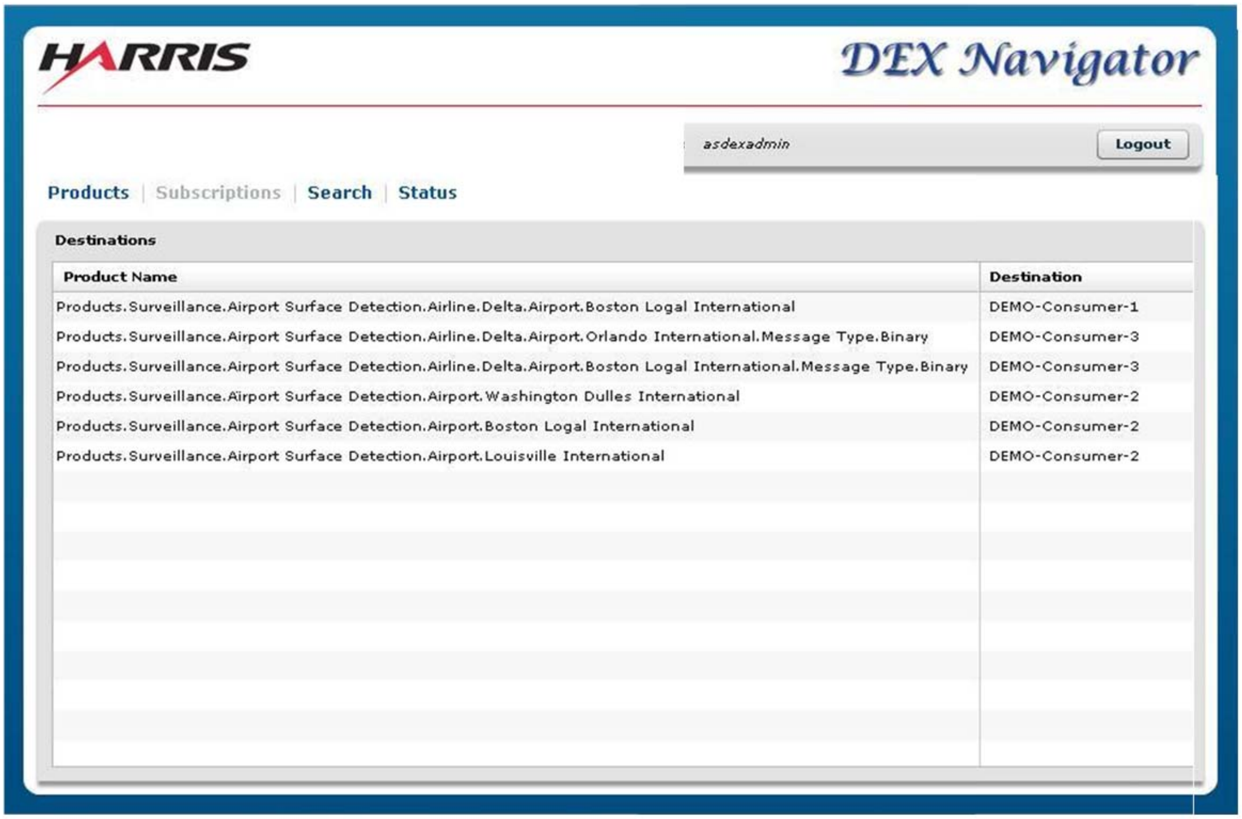


Figure 3-3. Example of Product Taxonomy[9]

The example in Figure 3-3 shows part of the taxonomy that has been defined for the ASDE-X service. The ASDE-X product taxonomy provides a structure in which each ASDE-X message is associated with a specific airport and some messages are associated with a specific airline. This allows consumers to select the messages they wish to receive based on the airline and airport that the message applies to. In the example shown, consumer system “DEMO-Consumer-1” is being subscribed to ASDE-X messages that refer to Delta flights at Boston Logan airport. With this subscription in place, the CBR function in the SWIM ESB will route all messages that refer to Delta flights at Boston Logan airport to that consumer system. Similarly, DEMO-Consumer-3 is subscribed to binary format messages pertaining to Delta airlines at Orlando and Boston, and DEMO-Consumer-2 is subscribed to all messages from Washington Dulles, Boston, and Louisville.

[9] This figure is a screenshot from the Data Exchange (DEX) Navigator, provided by Harris and used with permission.

When a producer publishes messages to the SWIM ESB using JMS, fields in the JMS message header are used to identify where the message falls in the product taxonomy. These fields are placed in the JMS header by the producer system, and used by the CBR function within the SWIM ESB.

The product taxonomy is defined during the producer on-ramping process. The initial set of consumer subscriptions is defined during the consumer on-ramping process, but may be modified thereafter as new consumers are added or as consumer needs change.

3.1.2.3 Publish/Subscribe with Very Large Products

Preliminary guidance from the NEMS service provider is that information products disseminated using the JMS publish/subscribe solution should be no larger than approximately 10 megabytes. Larger JMS product sizes can be supported, however the 10 megabytes maximum size is recommended to avoid potential performance problems. For larger products, a modified solution can be used in which the basic publish/subscribe solutions described above is used to allow the producer to notify consumers that a new product is available, but the product itself is obtained via some means other than the NEMS JMS publish/subscribe mechanism. For example, the product might be retrieved via a web service request/response service provided by the producer system. For brevity, further guidance on SWIM solutions for publish/subscribe for very large products is not included in this document, but is available on request from the SWIM program office.

3.1.2.4 NEMS Services Needed for Basic Publish/Subscribe Solution

To implement this solution, the producer program would need to order a “JMS-P Publishing Producer” POS. As part of the on-ramping process, the producer systems will be identified, and products to be produced and the “product taxonomy” would be defined.

The consumer program would need to order a “JMS-C Subscription Consumer” COS. As part of the on-ramping process, the consumer systems will be identified, and the products and sub-products that each consumer system is subscribed to will be determined using the product taxonomy.

The producer program would also order an Exchange Service associated with the dissemination of products from the producer systems to consumer systems.

3.2 Request/Response Use Case

In this use case, a user program needs to provide a means for a producer system to provide business services upon request.

3.2.1 Use Case Overview

An overview of this use case is provided in Figure 3-4.

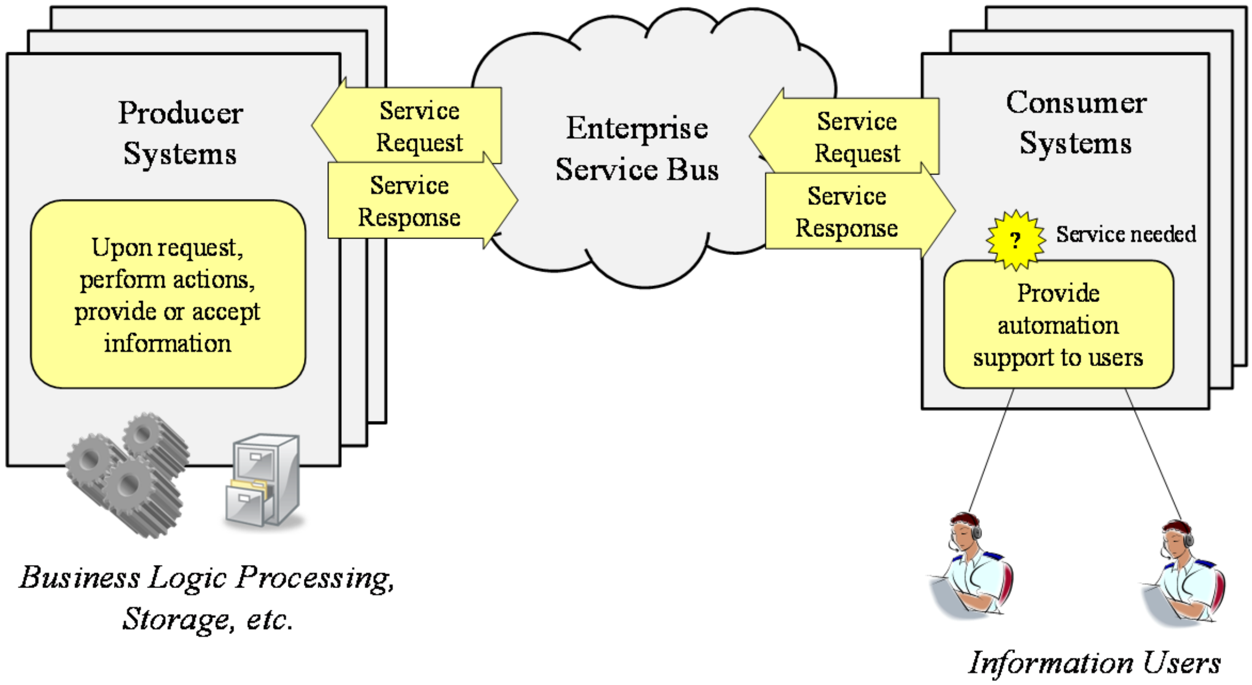


Figure 3-4. Overview of Request/Response Use Case

In this scenario, a consumer end system, in the course of providing automation support to human users, determines that some business service is needed from another system. The business service in question may provide data; it may accept data for processing, storage, or redistribution; or it may cause some other function to be performed (e.g., turn on runway lights, reroute a flight). The need is to provide a solution that allows the consumer system to make a request, to be acted upon by the producer system. The producer system will provide a response that may include data requested by the service consumer, status or results of processing, or a fault indication. In this use case, all requests for service are directed to the ESB, which is responsible for relaying the request from the service consumer to the service producer, and relaying the response back. The ESB also implements security functions, and can be used to provide enhanced availability and facilitate failure recovery, as well as other functions described in Section 4.

An example in which this use case might occur can be found in the concepts being developed for the Oceanic Conflict Advisory System (OCAS). The OCAS concept is to provide tactical trajectory feedback to oceanic airspace users to help their flight crews make informed clearance change requests to controllers. To implement this concept, OCAS could act as a SWIM producer system that makes aircraft and airspace conflict advisory information available as a business service. (This is a notional concept at this point.) Airline and Department of Defense (DoD) flight planners considering potential flight profile changes in Flight Operations Centers (FOCs) would use their automation systems to prepare and evaluate these flight profile changes. The FOC automation systems, acting as SWIM service consumers, would issue requests for the OCAS aircraft and airspace conflict advisory business services as needed, and use the results to meet the needs of the flight planners.

A key difference between this use case and the previously described "Publish/Subscribe" use case is that the service request can contain parameters that are specified by the consumer system at the time the service is invoked. For example, these parameters may specify the information products the consumer system needs at that time. (Contrast this with the previous use case, in which the products that each consumer should receive are defined in advance.) Also, in this use case the service consumer system controls when it wants to invoke the service. This is contrasted with the “Publish/Subscribe” use case, in which the products that each consumer is subscribed to are determined in advance, and the producer system controls when the products are provided, based on, for example, the availability of new data from sensors. Another difference is that in a Pub/Sub scenario, N consumers requiring data will be accomplished by only one data push from the producer.

3.2.2 Request/Response Solution

3.2.2.1 Solution Overview

An overview of this solution is shown in Figure 3-5.

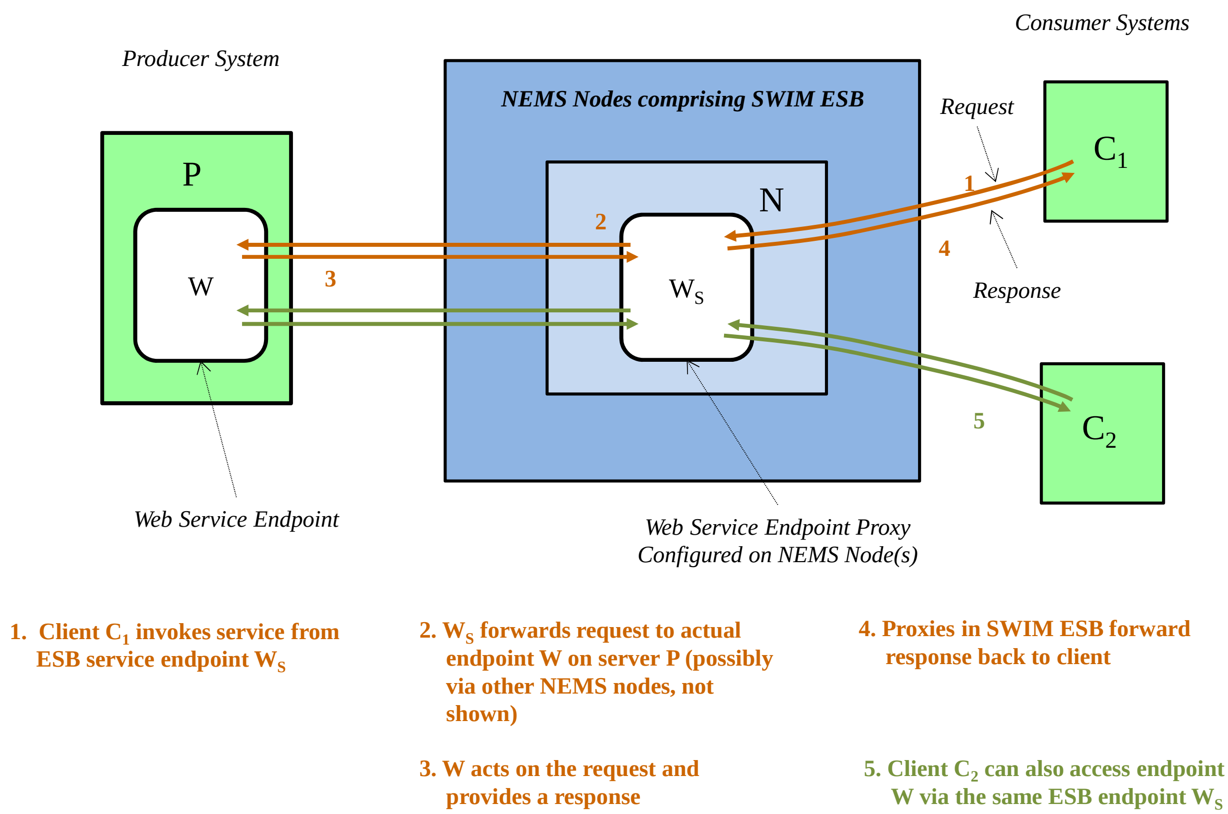


Figure 3-5. Overview of Request/Response Solution

Figure 3-5 shows a producer system[10], P, which provides a SWIM business service W. This business service is implemented as a web service that can be invoked via Hypertext Transfer Protocol (HTTP) in order to retrieve information, provide information, or invoke some

[10] A special note on terminology is in order. We are using the term “producer system” to refer to a system that provides a web service, and “consumer system” to refer to a system that invokes a web service. In common usage, a system that provides web services is often referred to as a “server” and a system that invokes web services is referred to as a “client”. We keep to the “producer/consumer” terminology for historical reasons and because it allows us to use the same terms as we are using for JMS solutions. However, the reader needs to bear in mind that a client can send information to a server using web services, thus with web services the flow of information may not always be what might be implied by the terms “producer” and “consumer”.

processing that implements a NAS function. Different styles of web service are supported by SWIM, including SOAP, Representational State Transfer (REST), and plain old XML (POX). The figure also shows two consumer systems, C_1 and C_2, which need to access the SWIM business service W. They do so by invoking a SWIM web service endpoint W_S located on a NEMS node N within the SWIM ESB[11]. W_S is a web service proxy that is instantiated by the SWIM program. The proxy W_S is configured to forward requests to the actual endpoint W on the producer system. Web service proxies maintain HTTP session state, so that when the endpoint W responds to a service quest, the proxy can forward the response back to the appropriate consumer, allowing different consumers to use the same proxy to invoke the same web service, as illustrated. Note that the consumer systems do not need to know that they are accessing a proxy – they simply access the business service at the location W_S that will be published in the NSRR; the SWIM ESB instantiates all required proxies and interaction to support the service.

The response returned by a web service would normally contain a message which provides the information product or results needed by the consumer. Alternately, the response may contain a reference (e.g. a URL) to location from which the product may be retrieved in a subsequent request.

As part of the producer on-ramping process, the producer systems and the web services they offer, all required service addresses, and the NEMS node they connect to, are explicitly identified. The consumer systems, and the web services they are authorized to access, are also explicitly identified during the consumer on-ramping process. The NEMS service provider is then responsible for configuring web service proxy endpoints as needed to allow requests and responses to be relayed between the consumers and producers.

3.2.2.2 NEMS Services Needed

To implement this solution, the producer program would use the “WS-P Web Service Producer” POS. As part of the on-ramping process, the producer systems and web service endpoints will be identified.

The consumer program would need to order the “WS-C Web Service Consumer” COS. As part of the on-ramping process, the consumer systems will be identified, and the web service endpoints that each consumer system is authorized to invoke will be determined.

The producer program would also order an Exchange Service for the dissemination of products from the producer systems to consumer systems.

3.3 Dynamic Subscription Use Case

In this use case, producers push data to consumers, based on a dynamic (run time) request in which the consumer specifies the products they want to consume.

3.3.1 Use Case Overview

An overview of this use case is provided in Figure 3-6.

[11] If necessary to efficiently transport requests or to cross security barriers (as discussed in Section 4) there may in fact be a chain of proxies configured in NEMS nodes in the ESB. Chained proxies are not shown in the figure, and if they are used this will be completely transparent to the consumers and the ultimate service provider.

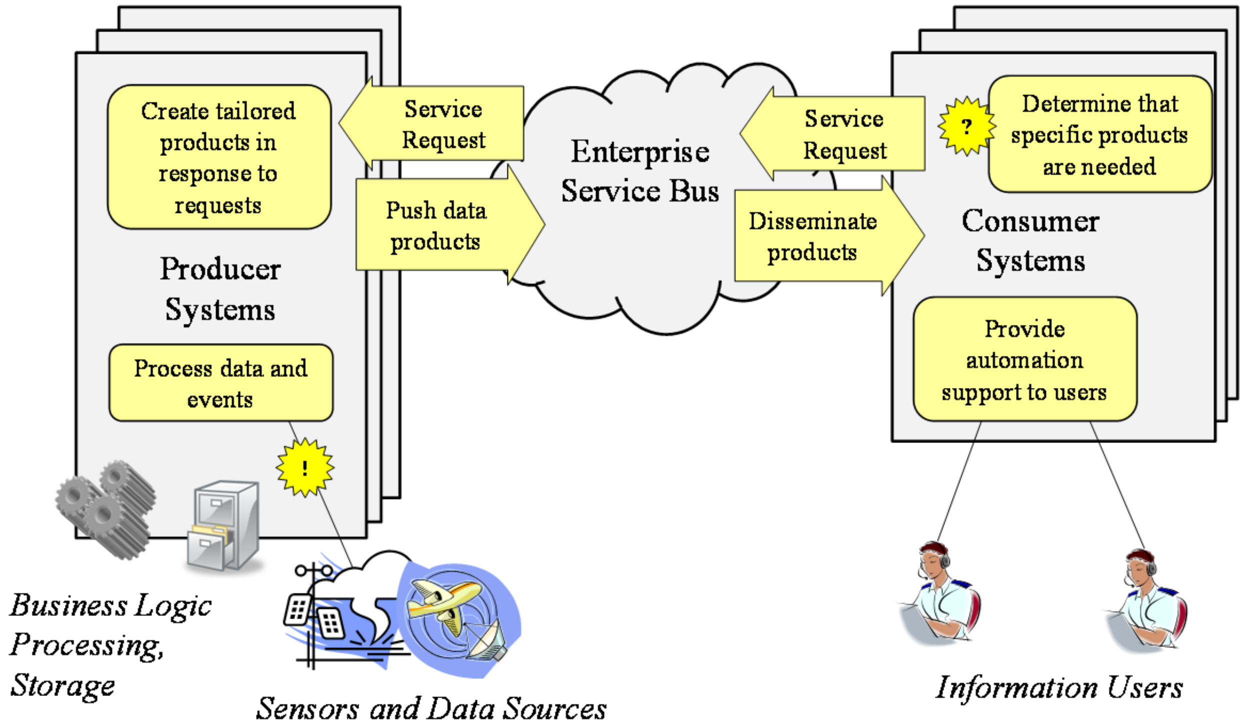


Figure 3-6. Dynamic Subscription Use Case Overview

This use case is essentially a hybrid of the previous use cases. Consumer systems, which are providing automation support to users, dynamically determine that they need to begin receiving a flow of information from another end system. When this happens, the consumer systems make a business service request, which is placed on the ESB. The service bus forwards the request to the appropriate producer system, which then begins to create products to meet the request, and pushes these products to the ESB. The ESB disseminates the products to the consumer system(s) that has requested them.

A notional example in which this use case might occur is in the monitoring of a national security event such as a gathering of world leaders at the United Nations. During such an event, automation systems at DoD and/or Department of Homeland Security (DHS) command centers may determine that they need to begin receiving track data for all aircraft within a certain volume of airspace. The DoD and DHS systems, acting as consumers, might issue a dynamic subscription request to an FAA producer system that provides track data as a SWIM business service. In response to the request, the producer system would begin publishing track data to those consumer systems, and continue until the consumer system cancels the request at the end of the event.

As in the request/response use case (Section 3.2) the information products that the consumer needs do not have to be known in advance; rather, they are determined by parameters specified in the service request. However, unlike the request/response use case, the result of the service request is not a single response; rather, the service request creates an ongoing flow of information from the producer to the consumer. This information flow may continue for some length of time, determined by the producer based on the availability of data or possibly based on parameters specified by the consumer in the service request.

3.3.2 Dynamic Subscription Solution

3.3.2.1 Solution Overview

The solution is implemented by a combination of web services and JMS, as illustrated in Figure 3-7.

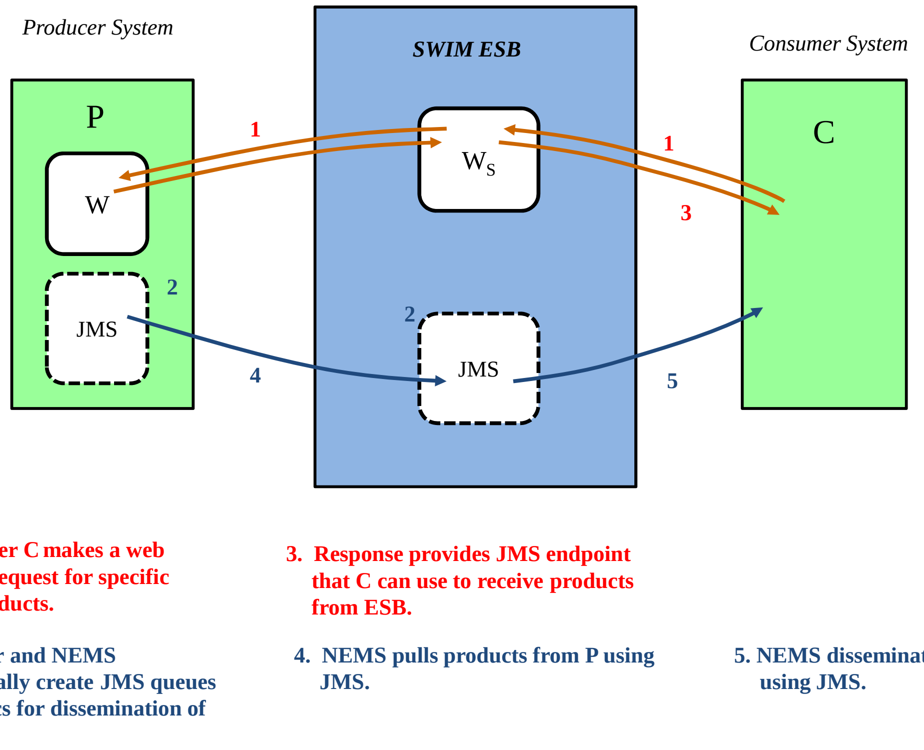


Figure 3-7. Dynamic Subscription Solution

The figure shows a consumer system C that needs to initiate a business service that will provide a flow of information from producer system P. System C invokes a service end point W_S on a NEMS node in the SWIM ESB to request this service. As in the request/response solution, W_S is a web service proxy instantiated by SWIM, which handles forwarding the web service request to end point W on the producer system. Upon receiving the request, the producer system P dynamically creates a JMS queue or topic for the products needed by C[12]. The producer system P then responds to C's web service request with address information for the new, dynamically created, JMS topic on the producer system. The SWIM ESB then dynamically creates new JMS queues and/or topics on NEMS nodes as needed for the dissemination of the new products from P to C. The web service response sent back to C provides the address of the JMS topic on the SWIM ESB that the consumer needs to connect to in order to receive its desired products. The producer system P then begins publishing data for consumer C to its local JMS queue or topic, and NEMS pulls the data into JMS queues and/or topics on the SWIM ESB, and finally disseminates the information to C, just as in the JMS publish/subscribe solution. When the dynamic subscription expires or is terminated by the consumer, the JMS connections are terminated and dynamically created JMS queues and topics are removed.

[12] For this solution, the producer needs to implement a JMS broker, also known as a "JMS Provider".

(For simplicity, the NEMS nodes within the ESB are not explicitly shown in Figure 3-7, but are used just as in the previous solutions.)

Just as with the basic Publish/Subscribe solution, for products larger than the recommended maximum JMS size, a modified solution can be used in which notifications are published, but the product itself is obtained via some means other than the NEMS JMS publish/subscribe mechanism.

3.3.2.2 NEMS Services Needed

The producer program would need to order the following services:

- "WS-P Web Service Producer" POS.
- "E8 Dynamic Subscription" enhanced service option.

The consumer program would need to order the following services:

- "WS-C Web Service Consumer" COS.
- "E8 Dynamic Subscription" enhanced service option.

The producer program would also need to order an Exchange Service. Only one Exchange Service is required no matter how many producer (and associated POS services) instances are instantiated.

4 Completing the Basic Solutions

In describing the basic solutions patterns in Section 3, many practical issues were omitted, in order to keep the explanation simple. In this section, we discuss how the various NEMS capabilities can be used to make the basic solution patterns into complete solutions suitable for operational use.

4.1 Security Solutions

This section describes how security solutions can be applied on top of the basic solution patterns. Topics discussed include authentication and authorization and NAS boundary protection. Security functions that protect the SWIM ESB infrastructure, but do not directly impact end systems, are considered internal SWIM ESB security functions and are not discussed here.

4.1.1 Authentication and Authorization

A fundamental goal of SWIM is that information should be available to *authorized* users wherever and whenever needed. Authentication and authorization are the security functions that allow access to be provided to authorized users, but denied to others. Authentication consists of assuring that the identity of an entity is known when that entity attempts to access SWIM to receive or provide information or access other business services. Authorization consists of determining which identified entities should have access to information or services, and which should not. The term “entity” in the preceding sentences may refer to an individual person, or a system or automated process. The latter is referred to as a “non-person entity”, or NPE. Producer systems and consumer systems are examples of NPEs.

The following sections provide, first, a description of the basic security mechanisms used for authentication and authorization, followed by descriptions of how these mechanisms are overlaid on the basic SWIM solutions that were described in Section 3 to meet different user program needs for authentication and authorization.

4.1.1.1 Authentication Building Blocks

First, we describe some of the basic mechanisms that are used by the SWIM ESB for authentication. The following sections will describe how these basic mechanisms can be used with SWIM publish/subscribe and request/response solutions. The basic mechanisms are:

- Username/Password Credentials.
- X.509 Certificates.
- Transport Layer Security (TLS).
- Web Service Security (WS-Security) Tokens.

4.1.1.1.1 Username/Password Credentials

NAS systems can be provided by SWIM with a username and password during the on-ramping process. These credentials can then be used to identify and authenticate the systems when the services and information products are accessed via the SWIM ESB.

4.1.1.1.2 X.509 Certificates

An X.509 certificate contains a “Distinguished Name” (DN) that identifies an entity, and a digital signature (or chain of signatures) that can be traced back to a certificate authority that is known and accepted by the SWIM ESB and by the end systems. The entities identified in the DN can be either individual people, or computer systems or applications, i.e., NPEs. As an example of the former, each individual FAA employee has an X.509 certificate loaded onto their FAA Personal Identity Verification (PIV) badge that can be used to authenticate the identity of that individual person. As an example of the latter, X.509 certificates are loaded onto end system web servers, and can be used to authenticate the identity of the server. In Segment 2a, SWIM is geared towards providing information exchanges among systems, thus we are concerned at present only with the latter case.

4.1.1.1.3 Transport Layer Security (TLS)

TLS is a protocol that can be used to create a secure connection between two endpoints over an IP network. In the case of SWIM, one endpoint would be an end system, and the other would be a NEMS node that is part of the SWIM ESB. TLS can provide a number of security functions, including supporting authentication by allowing secure exchange of username/password or X.509 based credentials that identify the systems at either end of a connection. TLS provides each system with a strong assurance that it knows the identity of the system to which it is connected, and that no “man in the middle” can intercept or interfere with the communications.

4.1.1.1.4 Web Service Security (WS-Security) Tokens

SWIM supports the use of SOAP, which is a protocol that provides an “envelope” for XML messages that can support various security functions, including the ability to carry “tokens” that support authentication and authorization.

4.1.1.2 Authentication and Authorization for Publish/Subscribe Solution

Figure 4-1 provides an overview of authentication and authorization applied to the publish/subscribe solution.

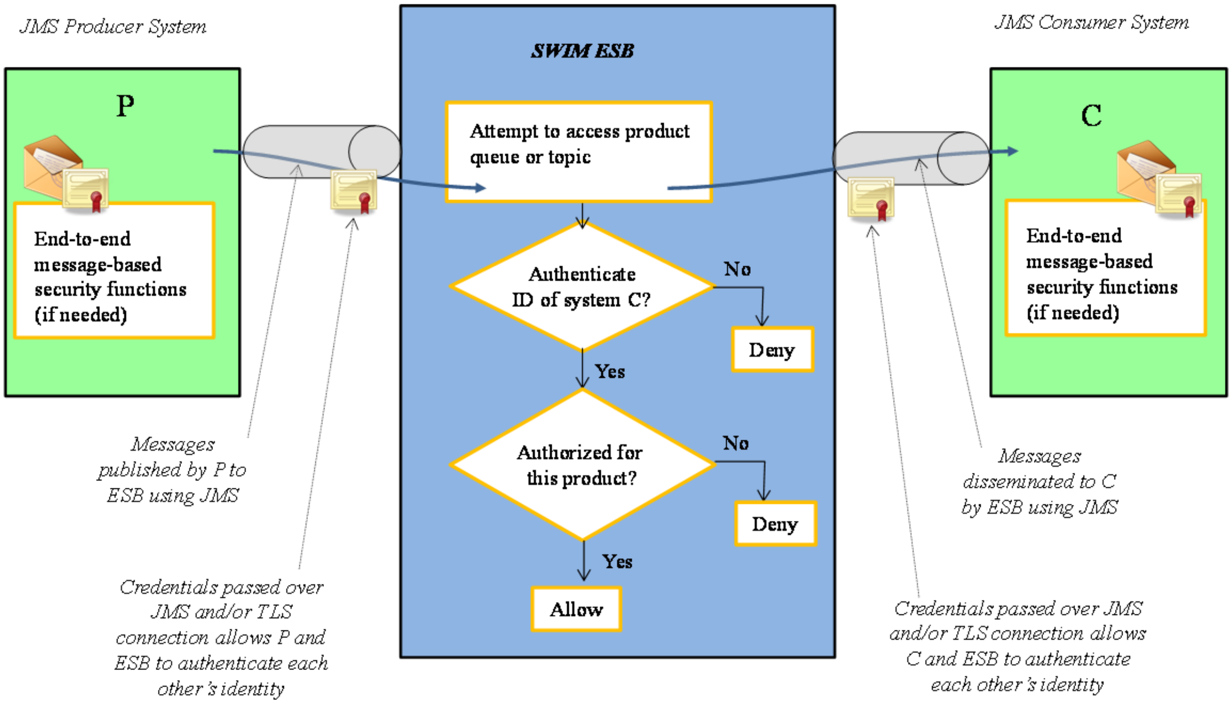


Figure 4-1. Overview of Authentication and Authorization Applied to Publish/Subscribe Solution

The identities of the end systems (producer systems "P" and consumer systems "C") are authenticated when they connect to the JMS broker at a NEMS node in order to attempt to publish products to a JMS queue or to subscribe to products from a JMS topic. Once the identities are authenticated, the NEMS nodes comprising the SWIM ESB will perform an authorization check to determine whether P is an approved publisher, and whether C is an approved consumer, for the product in question, based on the way P and C were configured during the on-ramping processes.

Authentication between end systems and NEMS nodes for JMS can be performed using the options described below.

4.1.1.2.1 Publish/Subscribe Authentication using JMS Username/Password.

The most basic approach to authentication for SWIM JMS publish/subscribe services is for the JMS client on the end system, when connecting to the NEMS JMS broker, to present the username and password assigned to that system during the on-ramping process. This allows the NEMS node to authenticate the producer and consumer end systems.

4.1.1.2.2 Publish/Subscribe Authentication using TLS and X.509 Certificates.

With this approach, when the end system connects to JMS broker on a NEMS node within the SWIM ESB, TLS is used to allow the end systems and NEMS to exchange X.509 certificates that can be used for authentication.

4.1.1.2.3 End-to-End Authentication Performed by End Systems

In addition to the options described above, if needed based on user program requirements, end-to-end security functions can be performed by end systems on a per-message basis. These functions would be implemented by the end systems, and this topic is not covered further in this document.

4.1.1.3 Authentication and Authorization for Request/Response Web Service Solution

Figure 4-2 provides an overview of authentication and authorization applied to the request/response web service solution.

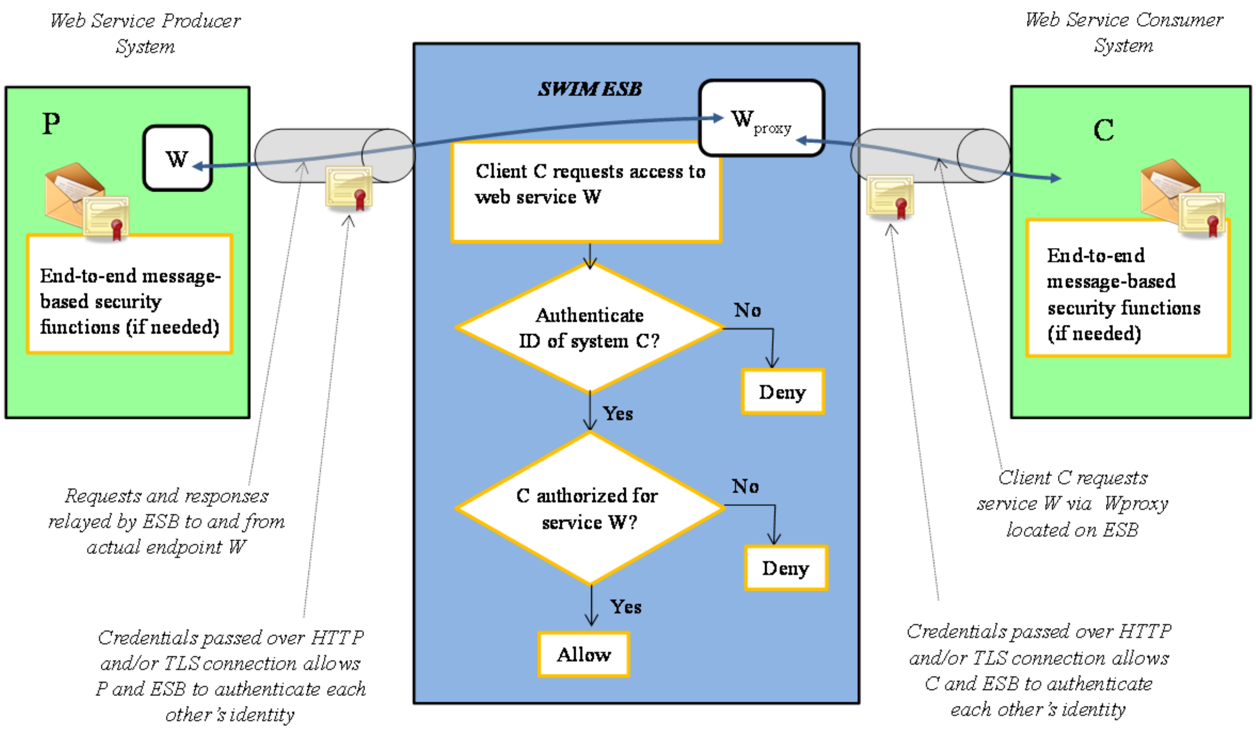


Figure 4-2. Authentication and Authorization Applied to Request/Response Web Service Solution

The figure shows a consumer system C accessing web service W provided by producer system P. The access is made via a proxy service W_{proxy} located on a NEMS node that is part of the SWIM ESB. (As discussed in Section 3, there may actually be a chain of proxies on NEMS nodes—however all proxy activity is transparent to P and C and is not shown in the figure.) As with the publish/subscribe scenario discussed above, the NEMS node will authenticate the identities of P and C, and perform an authorization check to determine whether system C should be permitted to access web service W, based on the consumer system profile and its role based authorization as defined in rules configured during the on-ramping process.

Authentication between end systems and NEMS nodes for Web Services can be performed using either of the options described below.

4.1.1.3.1 Web Service Authentication using WS-Security Token.

With this option, NEMS nodes will authenticate the identity of end systems using a token contained in the SOAP message envelope. SWIM supports web service authentication using the following type of token:

- WS-Security UsernameToken—carries Username/Password credentials.

- WS-Security BinarySecurityToken—carries X.509 credentials.
- Security Assertion Markup Language (SAML) Token—carries information about authentication, attributes, and authorization decisions that have been made by a trusted third party.

4.1.1.3.2 Web Service Authentication using TLS and X.509 Certificates

With this option, end systems and NEMS nodes may authenticate each other using X.509 certificates exchanged over TLS when an HTTP(S) session is established. This gives the end systems (C and P in Figure 4-2) a strong assurance that they are connected to NEMS, and it provides the NEMS node a strong assurance of the identity of C and P.

4.1.1.3.3 End-to-End Authentication Performed by End Systems

In addition to the options described above, if needed based on user program requirements, end-to-end security functions can be performed by end systems on a per-message basis. These functions would be implemented by the end systems, and this topic is not covered further in this document.

4.1.1.4 Authentication and Authorization for Dynamic Subscription Solution

As described in Section 3, the dynamic subscription solution uses a combination of the request/response web service solution and the JMS publish/subscribe solution, in which the JMS queues and topics are set up dynamically by the ESB as the result of a web service request from the subscriber. The authentication mechanisms used with the web services in the dynamic subscription solution are the same as the mechanisms described above for the request/response web services solution, and the authentication mechanisms used with JMS publish/subscribe services in the dynamic subscription solution are the same as the mechanisms described above for the JMS publish/subscribe solution.

The authorization rules for the dynamic subscription solution are defined during the on-ramping process. These rules will define which end systems are authorized to provide dynamic subscription services, and which end systems are authorized to request those services. SWIM will ensure that these rules are enforced by the NEMS nodes that make up the ESB, so that authorized end systems are allowed to access dynamic subscription services, and unauthorized end systems are denied access.

In addition to authorization performed by the ESB, if required, the producer system can also perform additional fine-grained authorization based on the identity of the system requesting the service and the parameters provided in the request.

4.1.1.5 Ordering Authentication and Authorization Services

Authentication using username and password credentials is provided when the services needed for the basic solutions presented in Section 3 are ordered. In addition, basic authorization functionality that verifies whether the producer and consumer systems are authorized to access the JMS products and sub products and web services are enacted when the services necessary for the basic solutions are instantiated. User roles and service access credentials are validated and put under configuration management as part of the NEMS on-ramping process; these credentials will be maintained for all authorized users and will be used for the authorization activities discussed earlier.

4.1.2 NAS Boundary Protection

In order to reduce the vulnerability of the NAS, FAA policy dictates that systems that are part of the NAS must not be connected directly to external systems or networks [12] [13]. However, NAS systems *do* need to exchange information with non-NAS systems such as those operated by airlines, DoD and DHS, and other FAA systems that are not considered part of the NAS. The NAS Boundary Protection Service (NBPS) is the service that makes this information exchange possible while keeping the NAS operational systems and networks isolated from external systems and networks. As illustrated in Figure 4-3, the NBPS has been enhanced to support SWIM services. That is, the NBPS has been updated to allow authorized external consumers to consume SWIM business services from the SWIM ESB and to allow authorized external producers to provide SWIM business services to the SWIM ESB. This document does not provide any internal details on how SWIM and the NBPS work together, since this information is sensitive and not needed by user programs. The figure is a high-level abstraction intended only to illustrate the fact that internal NAS systems (S_{int}) access the SWIM ESB by connecting to an internally visible NEMS node (N_{int}), and external systems (S_{ext}) access the SWIM ESB by connecting to gateway NEMS node N_{gw} that is accessible via the NESG[13]. The SWIM ESB and NBPS will provide the necessary boundary protection security controls to enable the necessary flow of messages and service requests and responses between the internal and external sides of the ESB.

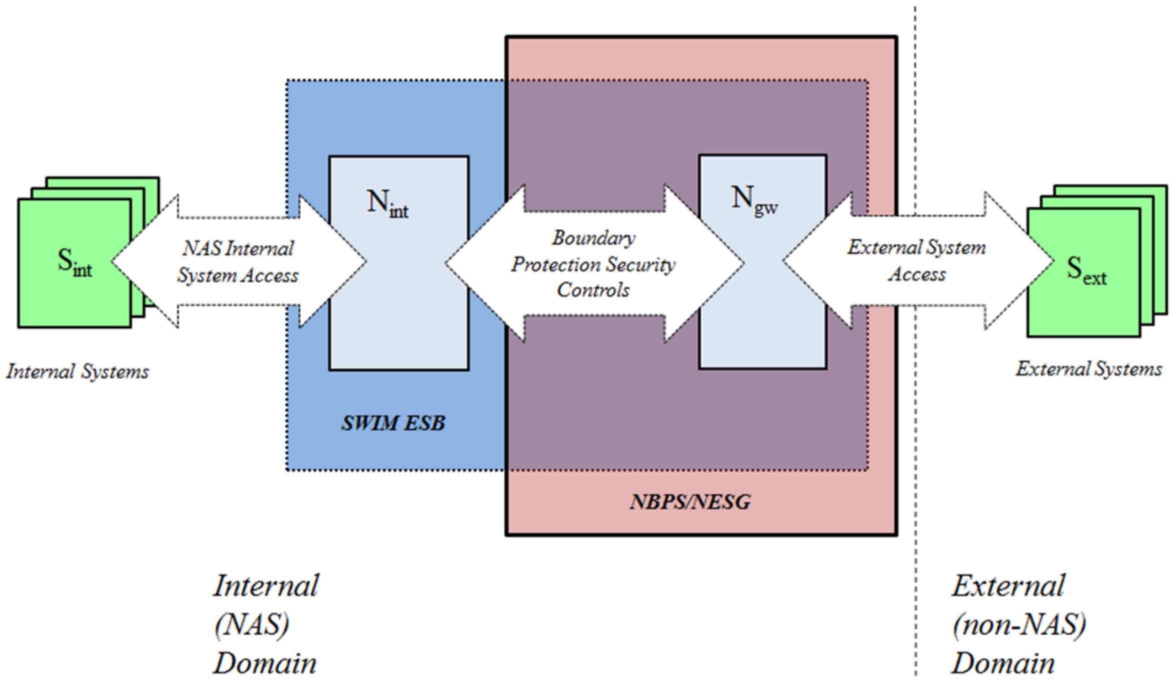


Figure 4-3. Overview of SWIM ESB and NBPS

The ability of the SWIM ESB to pass message content through the NBPS must be leveraged by NAS internal end systems (S_{int}) and external end systems (S_{ext}) that need to access each other's business services. User programs are required to leverage the NBPS for all IP-based interactions

[13] Connections between external systems and externally accessible NEMS nodes are made via the NESG. External systems must establish an encrypted dedicated circuit or encrypted VPN connection to the external interface at an NESG location. Further details are beyond the scope of this document.

between NAS and non-NAS entities. The SWIM program has worked closely with the NBPS provider and the NAS ISSM to put this solution in place to reduce the risk that external entities can attack NAS systems or networks.

The following paragraphs describe how boundary protection applies to the different basic solutions from Section 3.

4.1.2.1 NAS Boundary Protection for Publish/Subscribe Solution

Figure 4-4 provides an overview of NAS boundary protection applied to the JMS publish/subscribe solution.

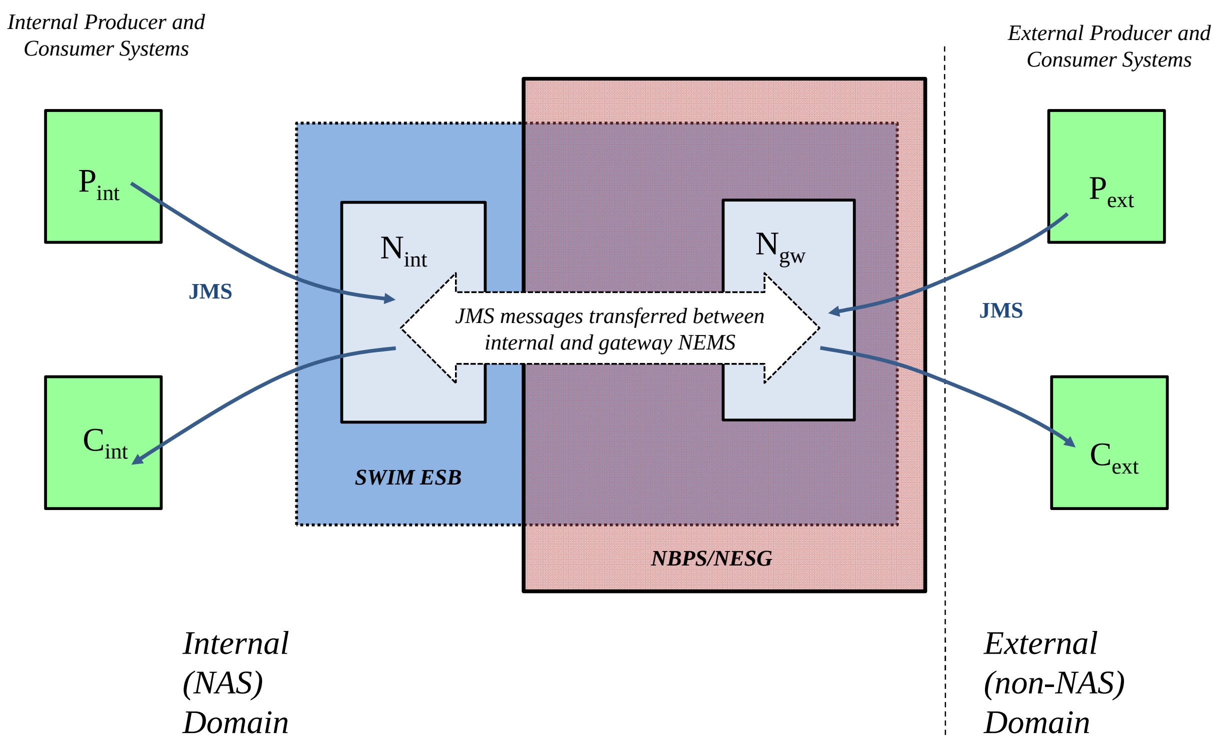


Figure 4-4. Overview of Boundary Protection for Publish/Subscribe Solution

Internal producer and consumer systems (P_{int} and C_{int}) access the SWIM ESB via an internal NEMS node (N_{int}) and external producer and consumer systems (P_{ext} and C_{ext}) access the SWIM ESB via a gateway NEMS node (N_{gw}), accessible via an encrypted connection to the NESG. Producers then use JMS to publish and consumers use JMS to subscribe, just as described in Section 3. The SWIM ESB will ensure that information flows between internal and external producers and consumers subject to boundary protection security controls and in accordance with the rules defined during the on-ramping process.

4.1.2.2 NAS Boundary Protection for Request/Response Web Services Solution

Figure 4-5 provides an overview of NAS boundary protection applied to the request/response web services solution.

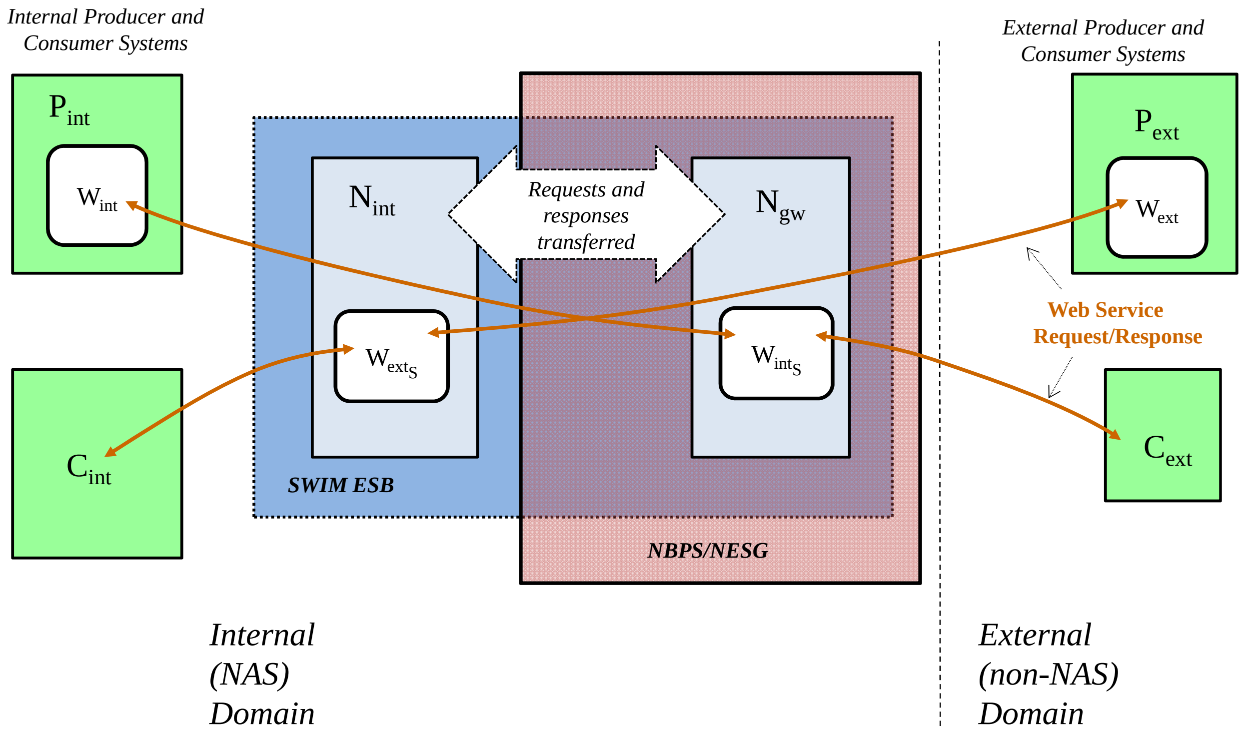


Figure 4-5. Overview of Boundary Protection for Request/Response Solution

The figure shows an external consumer system (C_{ext}) accessing a web service (W_{int}) provided by an internal producer system (P_{int}). C_{ext} cannot access W_{int} directly. Instead, access is provided via an externally accessible web service proxy (W_{intS}) within a NEMS node (N_{gw}) that is accessible using an encrypted connection to the NESG. The figure also shows the reverse flow: an internal consumer (C_{int}) accessing an external web service (W_{ext}) via a proxy (W_{extS}) on an internal NEMS node (N_{int}) within the SWIM ESB. Note that in fact there may be additional chained proxies beyond those shown in the figure; however, their presence is transparent to the end systems, so they are not shown or discussed further here. SWIM instantiates proxies as needed to forward requests and responses, and the NBPS allows these requests and responses to flow across the boundary subject to boundary protection security controls. In summary, a NAS system accesses an external business service via the SWIM ESB using interactions similar to those for NAS business services; the NBPS infrastructure supports all required security interactions. Likewise, a NAS producer provides a business service to external users via typical SWIM ESB interactions; again, the NBPS infrastructure supports all required security interactions to provision a NAS business service to external consumers.

4.1.2.3 NAS Boundary Protection for Dynamic Subscription Solution

Figure 4-6 provides an overview of NAS boundary protection applied to the dynamic subscription solution.

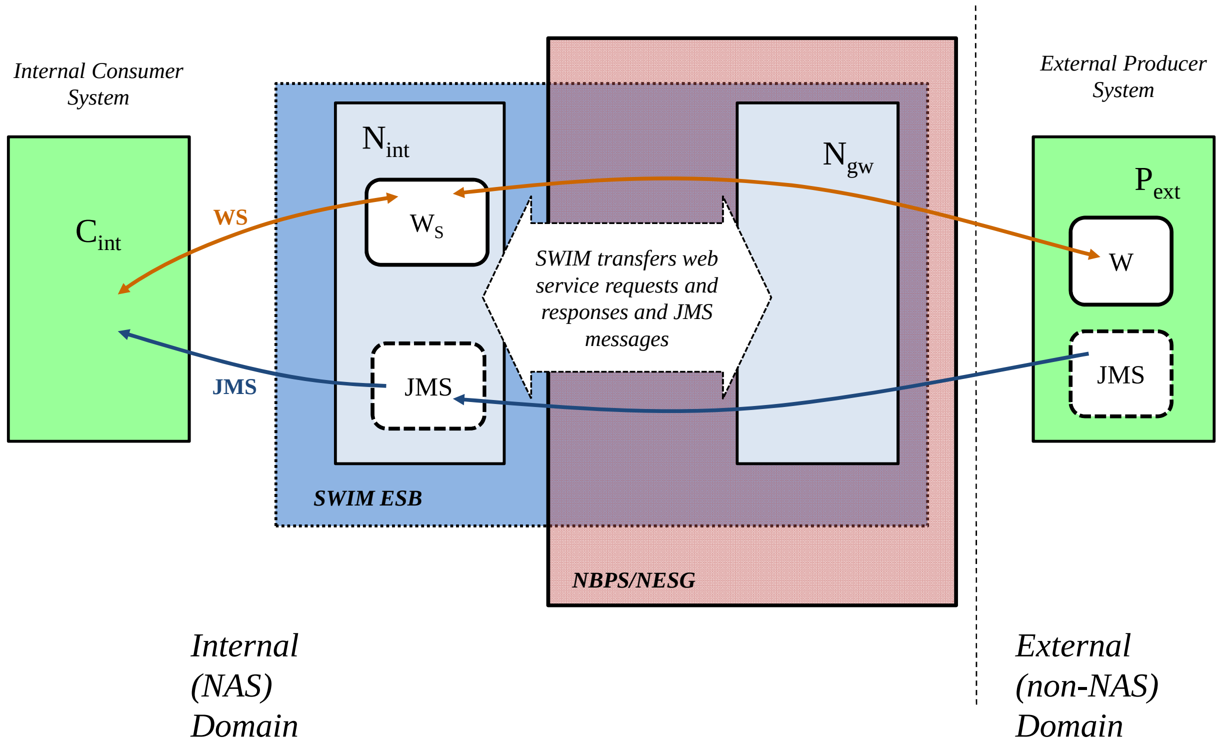


Figure 4-6. Overview of Boundary Protection for Dynamic Subscription Solution

As described in Section 3, the dynamic subscription solution uses a combination of the request/response web service solution and the JMS publish/subscribe solution, in which the JMS queues and topics are set up dynamically by the producer and the ESB as the result of a web service request from the subscriber. The figure shows the case where the information consumer system (C_{int}) is internal, and the producer system (P_{ext}) is external. In this case, the dynamic subscription web service request is made via a proxy (W_S) on an internal NEMS node (N_{int}). The SWIM ESB forwards the request via the NBPS to the endpoint W on the producer. The necessary JMS queues and topics are created on the producer system, and also on the NEMS nodes in the SWIM ESB. The SWIM ESB transfers JMS messages via the NBPS from the producer to a JMS topic on N_{int}. The consumer connects to the JMS topic on the N_{int} and retrieves the messages.

A similar configuration (not shown in the figure) can be used to allow the dynamic subscription solution to be applied if the producer is internal and the consumer is external.

4.1.2.4 Ordering Boundary Protection Services

Boundary protection services are included when the services needed for the basic solutions presented in Section 3 are ordered. Specific service classes and options have been established for external producer and consumer on-ramping services.

For more advanced security functionality, the programs building the producer and consumer systems would order the “E5 Security” enhanced service during the on-ramping process.

4.2 Availability and Failure Recovery

This section discusses availability features that are inherent in the NEMS infrastructure as well as the underlying IP network. A certain level of availability is provided to user programs with the basic producer and consumer on-ramping service. User programs that have requirements for higher availability can take advantage of several different failure recovery mechanisms. In this section, the availability numbers of the basic NEMS services are discussed. Next, possible ways to achieve higher level of availability based on orderable services are suggested. Other topics that relate to recovery of lost information are presented. Examples based on the use cases and solutions described in Section 3 are given throughout.

4.2.1 NEMS RMA Level Mapping and Availability Numbers

Table 4-1 shows RMA levels available for NEMS service categories. The mapping of RMA levels to orderable producer and consumer on-ramping services as well as Exchange Services are defined in the FTI Telecommunications Services Description (FTSD) document [14].

Table 4-1. NEMS Service Class Allocation and RMA Level Mapping [14]

Service Category	RMA Level
JMS Internal	RMA3
	RMA4
Web Services Internal	RMA3
	RMA4
JMS External	RMA3
	RMA4
Web Services External	RMA3
	RMA4

Table 4-2 provides restoration times for the NEMS services when ordered with the RMA levels shown in Table 4-1. The maximum restoration time for RMA level 3 is 8 minutes without diversity/redundancy restoration and 180 minutes with diversity/redundancy restoration. Diversity and redundancy restoration times are defined as the time required to restore full diversity and redundancy to a service [16]. RMA level 4 has the same maximum restoration time of 180 minutes.

Table 4-2. Restoration Times for RMA3 and RMA4 Service Levels [14]

Parameter Including Detection Time	RMA3	RMA4
Maximum Restoration Time (minutes)	8.00	180
Maximum Diversity/Redundancy Restoration Time (minutes)	180	180

Table 4-3 shows the minimum availability of NEMS services when ordered at each RMA level. Minimum availability is a calculated value that is determined over the latest 12-month period. Refer to [14] for a more detailed definition of RMA levels and details of the calculation of minimum availability.

Table 4-3. Minimum Availability for RMA3 and RMA4 Service Levels [14]

Parameter	RMA3	RMA4
Minimum Availability (based on latest 1 year period)	0.9998478	0.9979452

4.2.2 Overall Failover Context and Example

If the RMA levels described above are not sufficient to meet user program needs, additional mechanisms for recovering from various types of failures are available. An overall failover context is provided in this section, and more details will be provided in the following sections.

The overall context is depicted in Figure 4-7, which shows different failure recovery solutions overlaid on the basic publish/subscribe solution. (Similar mechanisms apply to the other basic solutions, and these are discussed in the text, but not included in the figure.)

Figure 4-7 shows how user programs can take advantage of dual RMA4 Ops IP layer services to maintain availability of NEMS services in the event of a network layer failure. In the example shown, by using dual RMA4 Ops IP services, producer P_1 has two separate paths to NEMS node N_{P1} and two separate paths to NEMS node N_{P2}. Similarly, consumer system C_n has dual paths to NEMS nodes N_{C1} and N_{C2}. In addition, dual paths are available between the NEMS nodes themselves. (The reasons for having the producer and consumer having paths to two different NEMS nodes will be discussed below.) With this architecture, no single failure at the network level can interrupt delivery of products from producer P_1 to consumer C_n. If the yellow path a from P_1 to N_{P1} fails, then the network will automatically switch over to the brown path a'. If the yellow path c fails, the network will automatically switch over to the brown path c', and so on.

Now we come to the reason why Figure 4-7 shows producer and consumer systems communicating with multiple different NEMS nodes. This allows NEMS services to be maintained even in the event that an entire NEMS node completely fails. As will be discussed in Section 4.2.6.1, the Global Load Balancers, working together with the NAS Domain Name System (DNS), can manage failover between NEMS nodes. In the example, if N_{P1} fails or is completely disconnected from the network, producer P_1 will be reconnected to N_{P2}, which will take over dissemination of products from producer P_1 to consumer C_n. Similarly, if N_{C2} fails, then C_2 will be reconnected to N_{C1}. This load balancing activity is a NEMS infrastructure internal activity and requires no service option be ordered by the NEMS users.

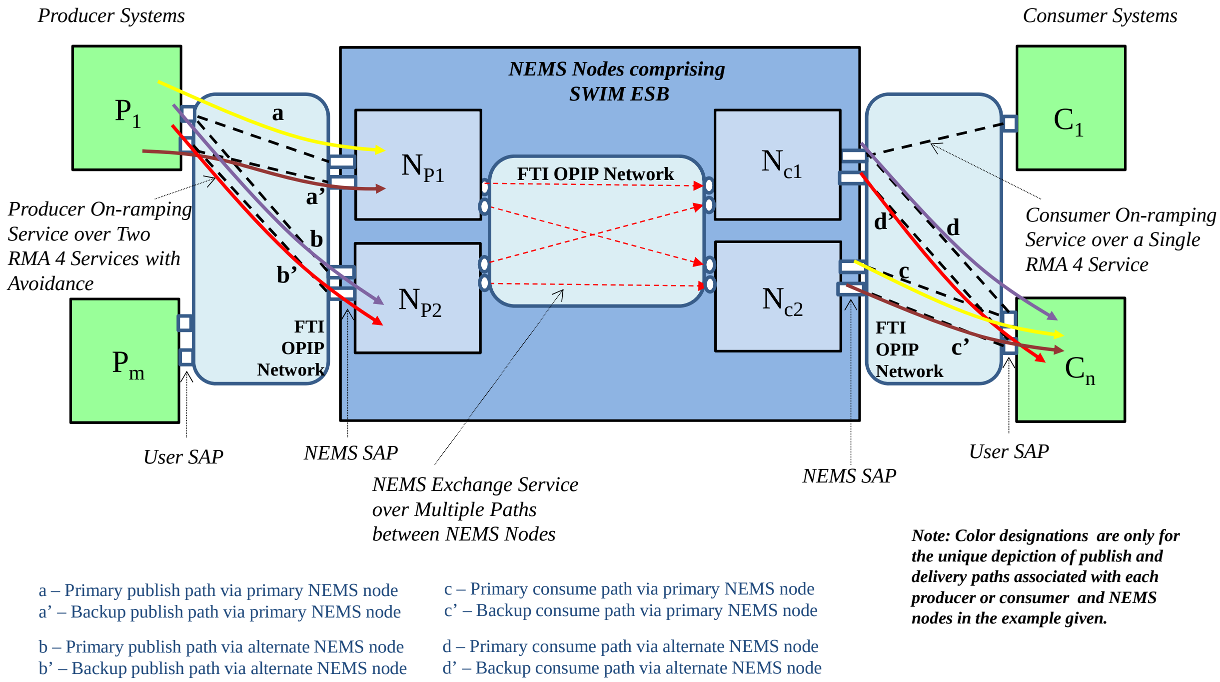


Figure 4-7. Failover and Enhanced Availability Mechanisms in NEMS

If the network failure is on the consumer side then JMS durable subscription can be employed to resume the product delivery process once the consumer is reachable again. Durable subscription, reconstitution, and reliable messaging are discussed in Sections 4.2.4, 4.2.5, and 4.2.7 respectively. Failover mechanisms for producers, consumers, and NEMS nodes are discussed in Section 4.2.6.

4.2.3 Enhancing IP Network Layer Availability

If the RMA levels shown in section 4.2.1 are not sufficient for user program requirements, both producers and consumers could increase availability by ordering a dual RMA 4 service with avoidance shown in Figure 4-8. The figure shows an end system connected to a NEMS node with both a primary and a secondary path over the FTI IP network. Whether the NEMS node is local or remote is transparent to the end system. Details of this service can be found in the FTSD as well as the *FTI Operational IP User's Guide* [8]. The service is discussed here because it is an underlying FTI capability that can be leveraged to achieve higher availability levels.

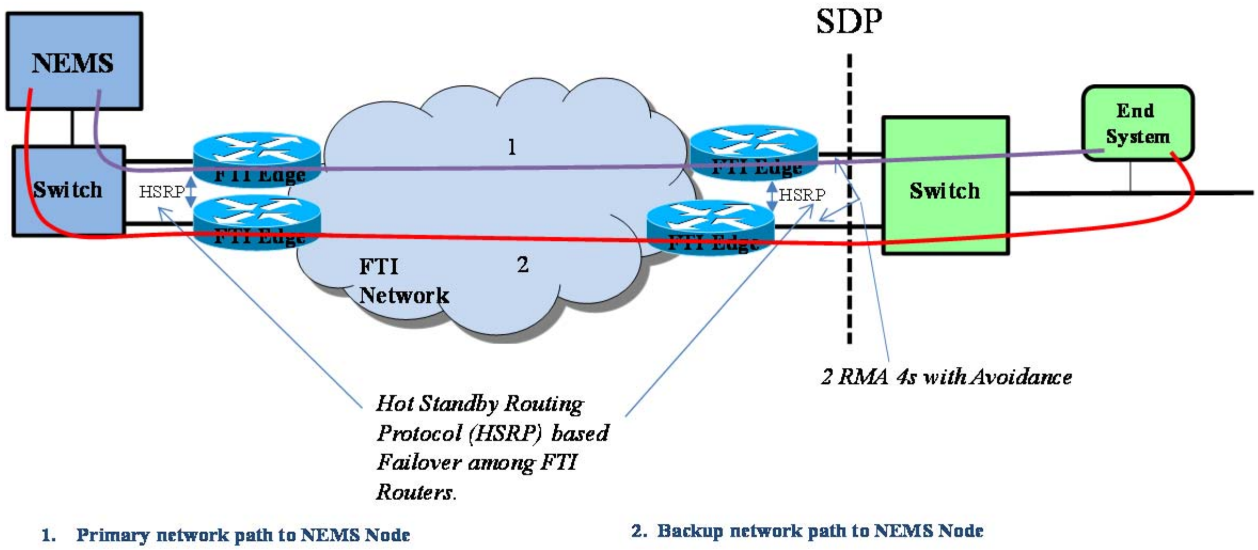


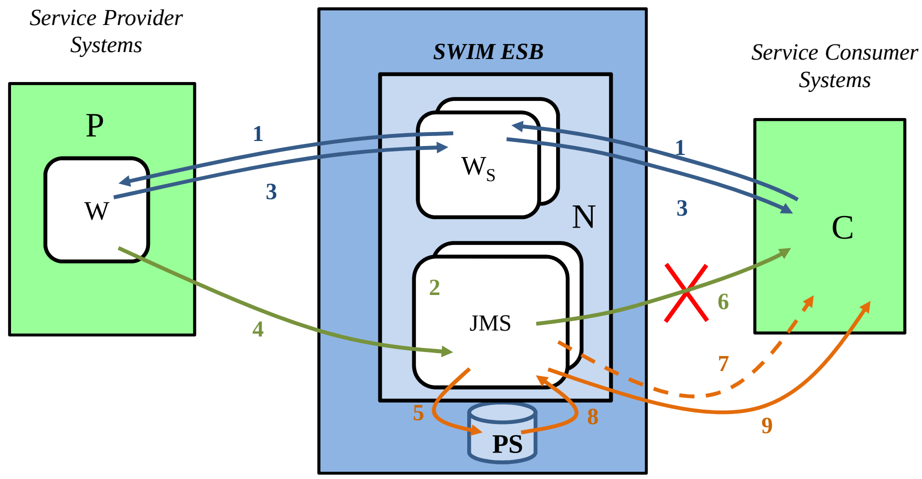
Figure 4-8. NEMS and User Interfaces Implemented via Dual RMA 4 Ops IP Services

4.2.4 Durable Subscription

Durable subscriptions enable consumers to recover subscribed content after a planned or unplanned disruption of consumer service or disconnection while publication is occurring. Once service resumes, consumers that have the durable subscription service are able to get any missed content from topics they have subscribed to upon resumption or re-establishment of service. This capability, which is available for NEMS JMS consumers, pushes the missed content to the consumers. The maximum disconnection period supported has direct dependency on the size of data storage available for persisting messages on nodes and is configurable per NEMS consumer topic. Secondary data storage is needed for a NEMS node to support durable subscription.

Currently, the SWIM program considers the outage window supported on a case-by-case basis during the on-ramping process.

Application of durable subscription to the basic publish/subscribe solution is shown in Figure 4-9. Durable subscription requires that messages be "persisted", i.e., written to secondary storage. With this service, if consumer C fails or is temporarily disconnected from the network, the messages that C has subscribed to will be saved until C is restored to operation or reconnected to the network. When C has been restored, messages are retrieved from storage and then delivered to the consumer.



1. C makes a web service request for specific data products.
2. NEMS dynamically creates JMS queues and topics for dissemination of products from P to C.
3. Response provides JMS endpoint that C can use to receive products.
4. P pushes products to NEMS using JMS.
5. Data is persisted in secondary storage PS.
6. NEMS node N dissemination of products to C is disrupted.
7. Connection is re-established between N and consumer C
8. N retrieves data from S that C was not able to get because of the disruption
9. Missed products are delivered to C

Figure 4-9. Publish/Subscribe Solution with Durable Subscription—Consumer Disruption

4.2.5 Reconstitution

Reconstitution is the process of resynchronizing or re-initialization of a given system's database based on up-to-date information in order to get to a complete and current picture of a certain NAS mission area. Such a need could arise due to an outage or disruption that forces operators of systems to re-initialize their database. The role of NEMS in accommodating reconstitution requests is limited at this time. NEMS will facilitate information exchange between producers and consumers but does not play an active role such as rendering and packaging reconstitution information unless that role was inherent in the business definition. Figure 4-10 shows an example of the role NEMS would play in reconstitution related message exchange between producers and consumers. In the figure consumer C, which may have just been reinitialized, invokes service W, which is a reconstitution service owned by the producer system P. The reconstitution service W is invoked via W_S, which is deployed on a NEMS node. W_S forwards the reconstitution request to the actual service endpoint W on server P. W acts on the request and invokes its business service that retrieves the requested reconstitution data from a single or multiple databases at its disposal. Once the appropriate information is retrieved, W responds back to C, via W_S, with the information needed to initialize C's database.

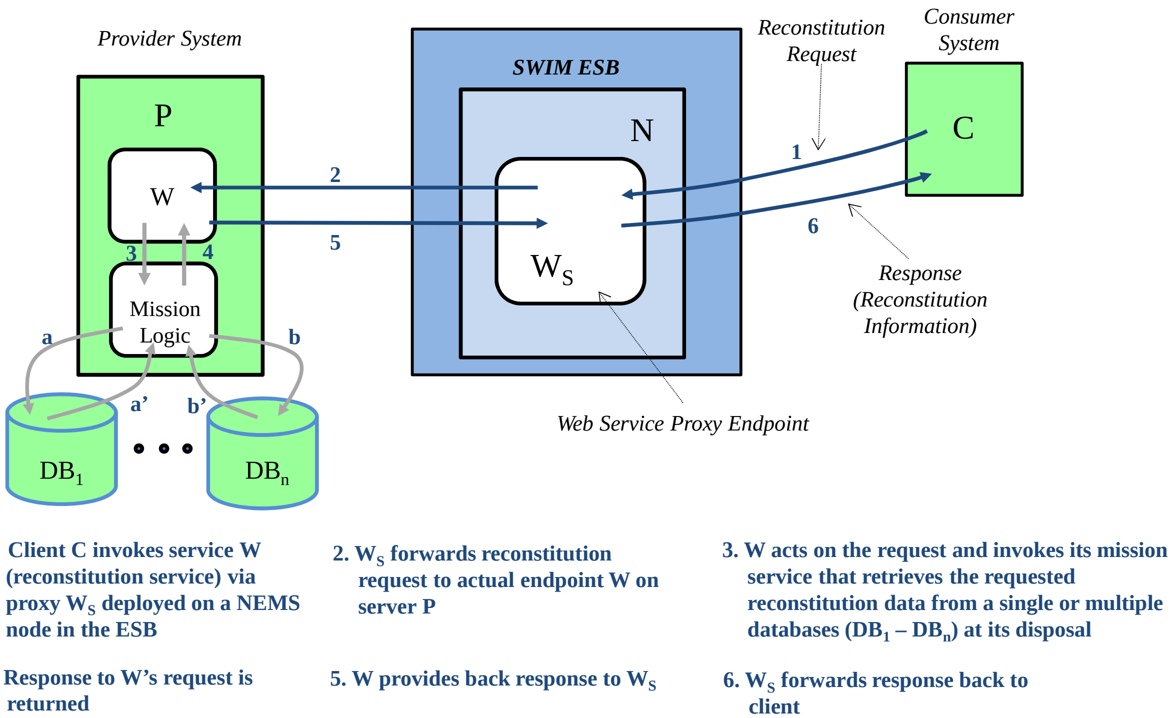


Figure 4-10. Request/Response Solution—Reconstitution

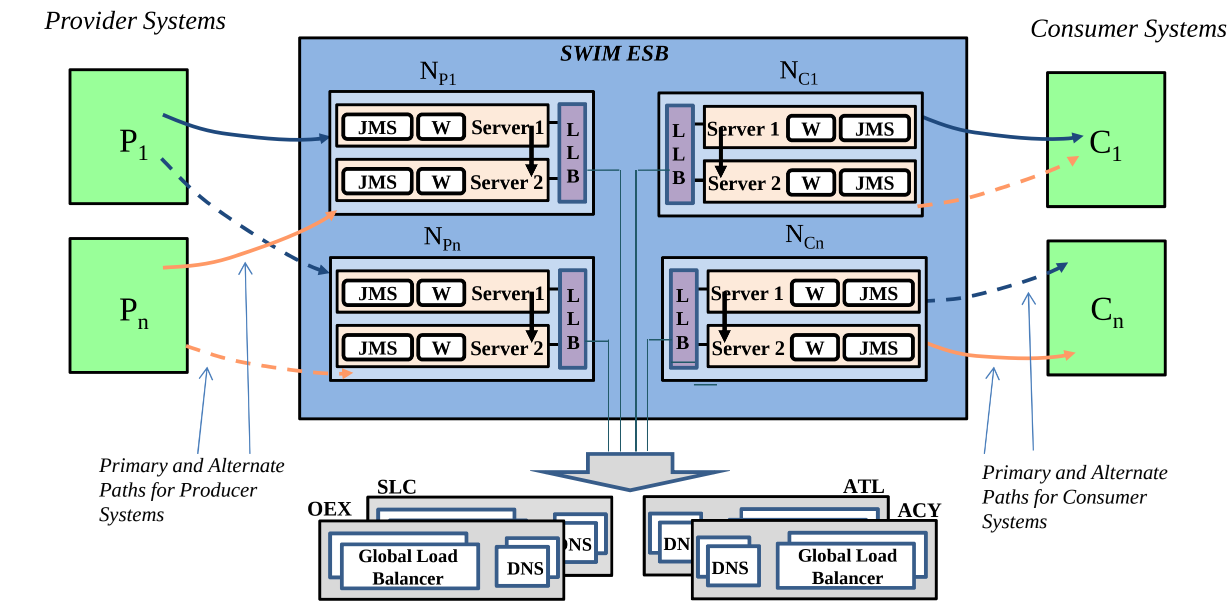
4.2.6 Failover

This section describes mechanisms for recovering from NEMS failures. In addition, guidance on producer and consumer system failover mechanism will be provided where NEMS may not be an active participant.

4.2.6.1 Global and Local Load Balancers

NEMS traffic among multiple geographically distributed SWIM ESB sites is load balanced across the NAS using Global Load Balancers. Furthermore, Load Balancers communicate with NEMS nodes at each site so that they can determine whether any given NEMS node has failed. If a NEMS node fails, the Global Load Balancers update DNS so that producer and consumer systems will be redirected to alternate NEMS nodes located at different sites. Redirection occurs for specific producers and consumers based on configured policies. There are several Global Load Balancers in the NAS.

Global Load Balancers are able to obtain the health and status information of the NEMS nodes via the Local Load Balancers that are located at all NEMS node sites. Local Load Balancers balance traffic among local server pools associated with a given NEMS node and provide performance optimization resulting from effective resource utilization of NEMS servers. In the event of an individual server failure within a NEMS node the Local Load Balancer will select an alternate local NEMS server.

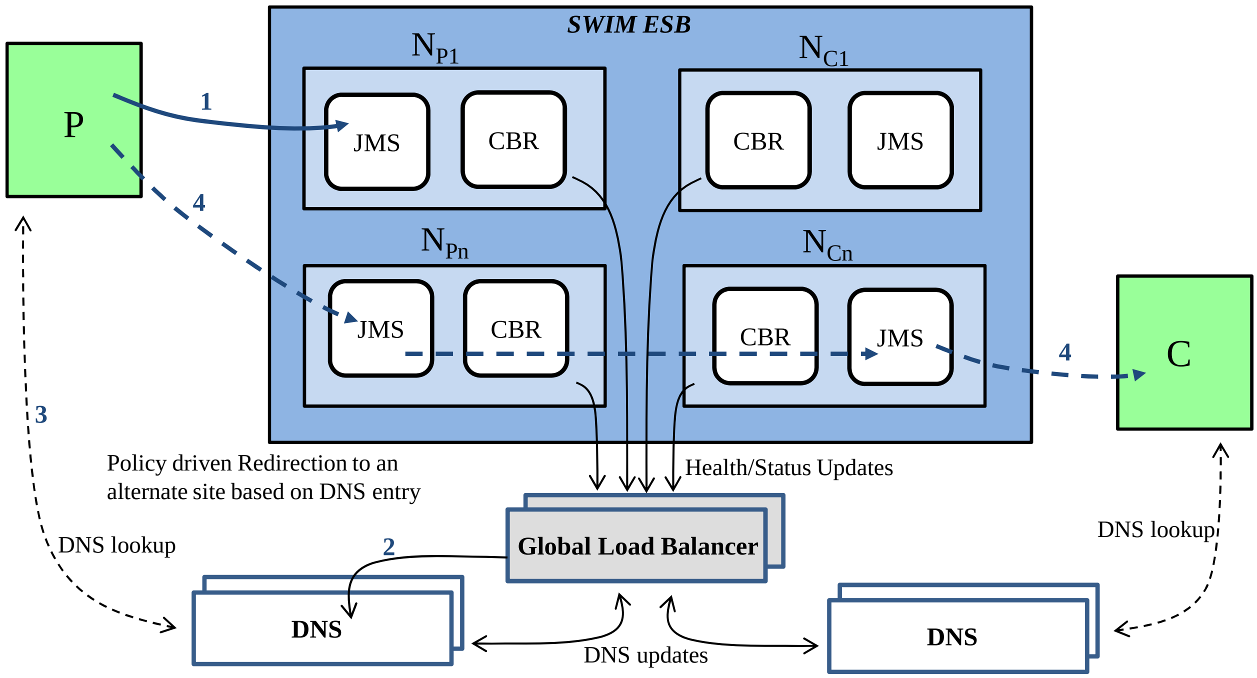


LLB – Local Load Balancer
DNS – Domain Name Service
SLC – Salt Lake City
OEX – Oklahoma City
ATL – Atlanta
ACY – Atlantic City

Figure 4-11. Global and Local Load Balancers

Figure 4-11 shows Global and Local Load Balancers relative to a NEMS node. In the figure, the Local Load Balancers are shown managing failovers between server blades 1 and 2 while also interacting with the Global Load Balancers. If a failure occurs within a NEMS node, the Local Load Balancer will transparently redirect producers and consumers to a different internal server blade. If persistence has been enabled and reliable messaging services are being used (see section 4.2.7), then messages that were "in transit" within the NEMS node at the time of failure will not be lost. The persistent stores are shared by the servers within a NEMS node, so that when a server fails a new server will take over processing of messages in the persistent store.

Figure 4-12 shows a failover scenario where a Global Load Balancer determines failure of a NEMS node N_{P1} and redirects producer P_1 to publish to an alternate NEMS node. Note that in this scenario in which an entire NEMS node completely fails, even if persistence had been enabled within node N_{P1}, messages that were stored there when the node failed could be lost, since persistent stores are not mirrored across different NEMS nodes.



1. **Producer P attempts to initiate publishing to Producer Node Site 1 (N_{p1}) which is the default assigned NEMS node for P**

2. **In the meantime the Global Load Balancer updates DNS with the address of an alternate node (N_{pn}) due to failure of Np1.**

3. **DNS updates its table and when P does a DNS lookup because it does not get a response from N_{P1}, it will get the address of the alternate NEMS node Npn.**

4. **P publishes to the alternate node N_{pn} and product is forwarded to N_{cn} and delivered to C .**

Figure 4-12. NEMS Node Failover Based on Global Load Balancer Updates to DNS

4.2.6.2 Use of Alternate Path Logic on End Systems

It is possible to implement application level logic on end systems which can be used for failover in the event that a given producer system fails. An example would be a case where there are multiple producer systems, each in a different NAS facility, and a given external consumer system accesses the producer's business services via web proxies on NEMS. Simple application level logic (URL selection) could be implemented on the consumer system to point to an alternate provider system in the event the primary one fails. In such a situation, NEMS does not play a role other than facilitating message exchange. Figure 4-13 shows this example.

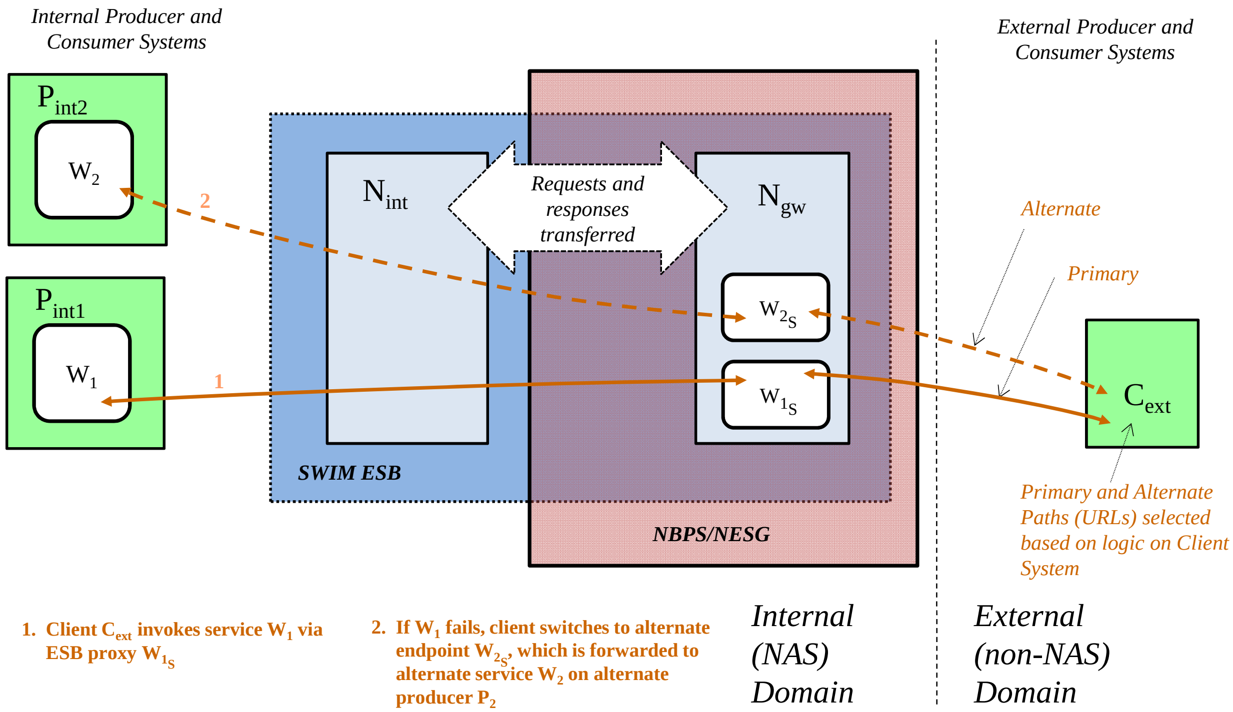


Figure 4-13. Alternate Provider System Selection Based on Logic on Consumer Systems

4.2.7 Messaging Reliability

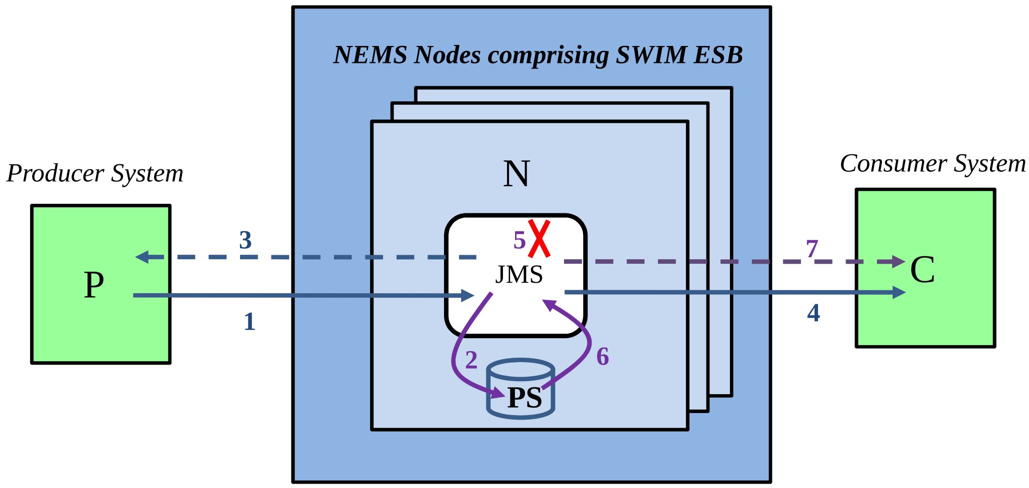
The NEMS Enhanced Service 2 (E2) capability[14] provides guaranteed message delivery for JMS and WS consumers.

For JMS there are two levels of guaranteed delivery supported. They are:

- At least once: message will be delivered, duplicates are possible.
- Once-and-only-once: message will be delivered, duplicates are precluded.

Figure 4-14 shows JMS based guaranteed delivery service that leverages persistence to provide guaranteed delivery even if a NEMS node needs to recover from an internal failure.

[14] The durable subscription service discussed earlier in Section 4.2.1 comes under the umbrella of the E2 service, and so does persistence.



1. Producers publish products to NEMS

2. Message is persisted in persistent storage PS

3. Acknowledgement for successful delivery from NEMS to Producer P

4. Message is pulled out of queue and delivery to C begins

5. While step 4 is taking place the JMS provider has a failure (after message is taken off queue) for a brief period and recovers

6. Message is retrieved from PS

7. In this example “at least once” delivery to C takes place successfully with acknowledgement from Consumer C via its JMS client

Figure 4-14. JMS Based Guaranteed Delivery

In addition to the JMS based guaranteed delivery service, NEMS also supports Web Services Reliable Messaging (WS-RM). The WS-RM standard specifies a number of delivery assurances similar, but not identical, to those provided in the JMS world, supported under the NEMS E2 service.

4.2.8 Priority

NEMS supports priority based messaging for JMS producers and consumers. For NEMS priority based messaging, the highest priority message will be removed from a queue and processed before lower priority messages are processed. It is important to note that a JMS provider tries to deliver higher-priority messages before lower-priority ones but does not have to deliver messages in exact order of priority. Therefore, it is important to note that although message priority is supported by NEMS, it is not guaranteed.

4.2.9 In-Order Message Delivery

In-order message delivery is not guaranteed. If order is important for a NAS application some mechanism such as sequence number or time tag would have to be put in the header of the message by the producer so that the consumer can deal with out-of-order delivery of messages.

4.3 Latency

There is a requirement for the NEMS Exchange Service to provide a minimum level of latency for its messaging services. This requirement, stated in [14], specifies that in the context of NEMS message exchange, the latency of messages traversing the SWIM ESB is defined as the time taken for a message to traverse from the source NEMS SAP to the destination NEMS SAP, as seen in Figure 4-15.

The latency requirement as defined in the FAA Telecommunications Specification Document (FTSD) stipulates that it be less than or equal to 1000 milliseconds (ms), for a message size of 500 bytes.

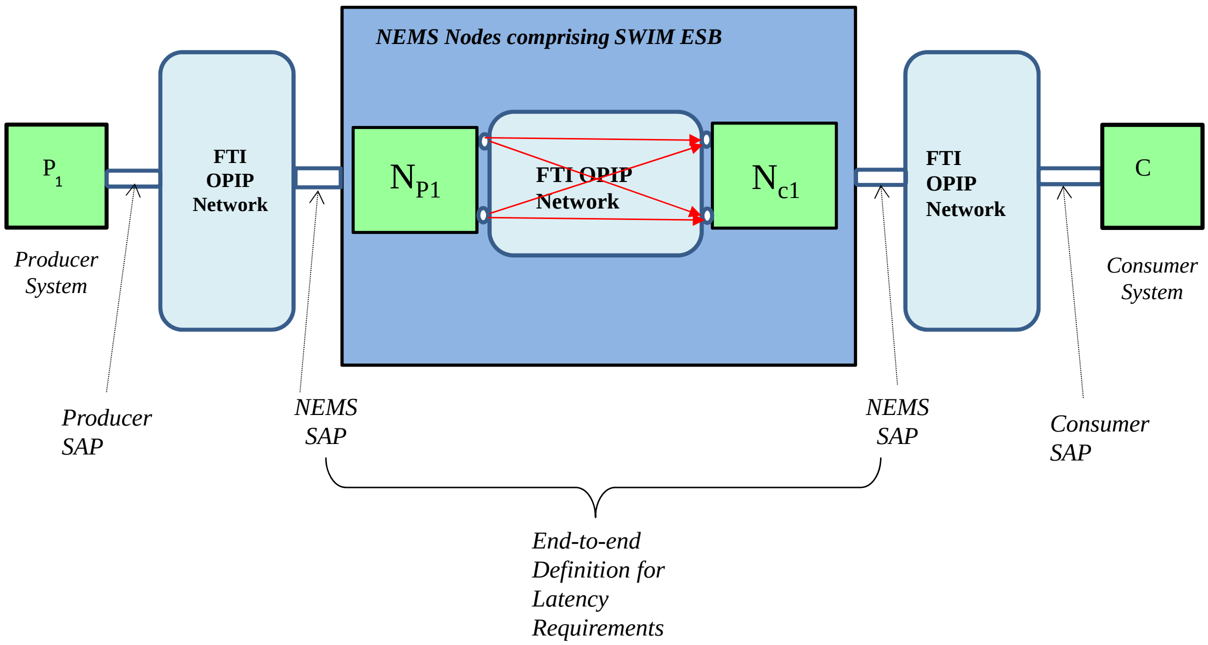


Figure 4-15. Latency Requirement Applies from NEMS SAP to NEMS SAP

4.4 Mediation

The NEMS mediation capability (E1 service) provides message content and protocol transformation. Message content transformation entails reformatting or modifying message payload content while protocol transformation involves communicating with a producer using a certain protocol and communicating with a consumer using a different protocol. For example, a producer could use web services to on-ramp information onto NEMS, but a consumer could get the same information via JMS publish/subscribe service as a result of mediation that takes place on NEMS. Extensible Stylesheet Language Transformations (XSLT) and XQuery are used to specify and perform transformations on textual message payloads. The following sections show use cases and some examples of mediation.

4.4.1 Protocol and Message Format Transformation

Protocol transformation enables disparate producers and consumers to communicate without having to modify either producer or consumer software. The SWIM ESB can be configured to provide the required protocol transformations for JMS and/or WS producers and consumers. The SWIM ESB can support multiple different implementations of JMS. The currently supported implementations are WebLogic JMS and ActiveMQ. WebLogic JMS is the recommended broker unless there are special circumstances where ActiveMQ is already in use by producers and consumers. The mediation function provides interoperability between these different JMS implementations.

Figure 4-16 shows an example of protocol mediation between Web Services and JMS. The producer uses web services to get information onto NEMS where transformation occurs so that the information can be published to a JMS topic queue for delivery to a JMS consumer.

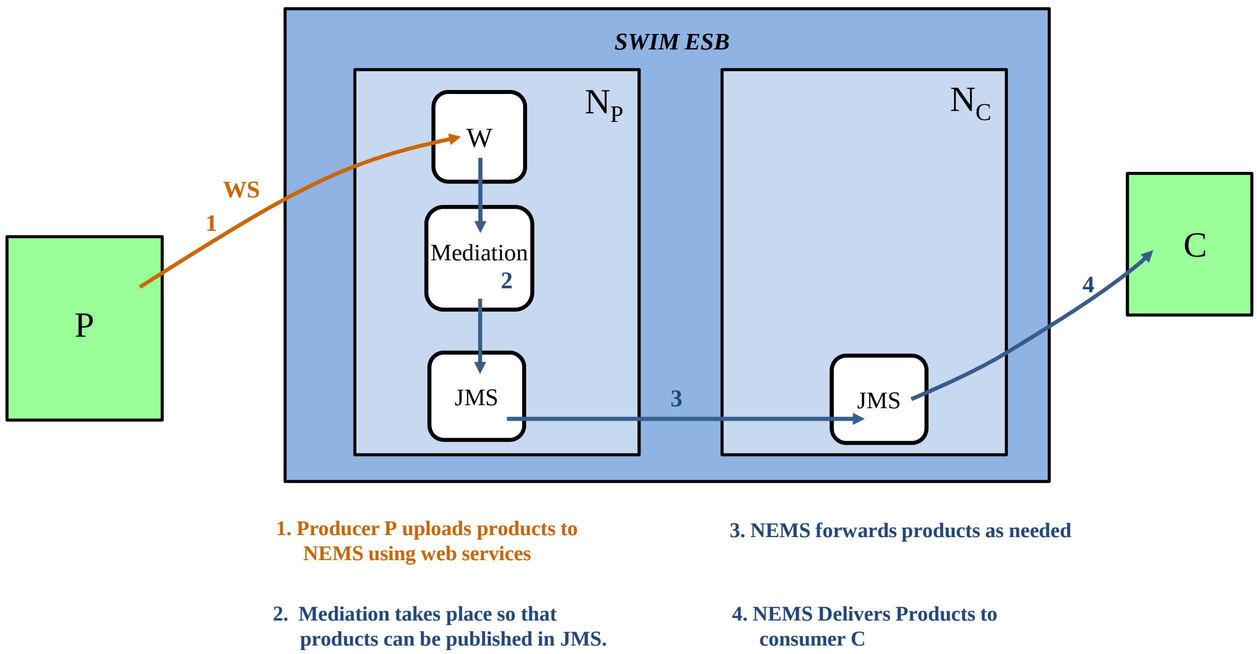


Figure 4-16. Mediation between WS and JMS

For message format transformation, a configurable set of rules defined using XSLT and/or XQuery are used to reformat or change message payload content. Transformation of non-XML to XML formatted messages can take place using XQuery. Transformation between different XML representations can take place using XSLT or XQuery. Figure 4-17 shows such a transformation where a consumer and producer use different XML formats.

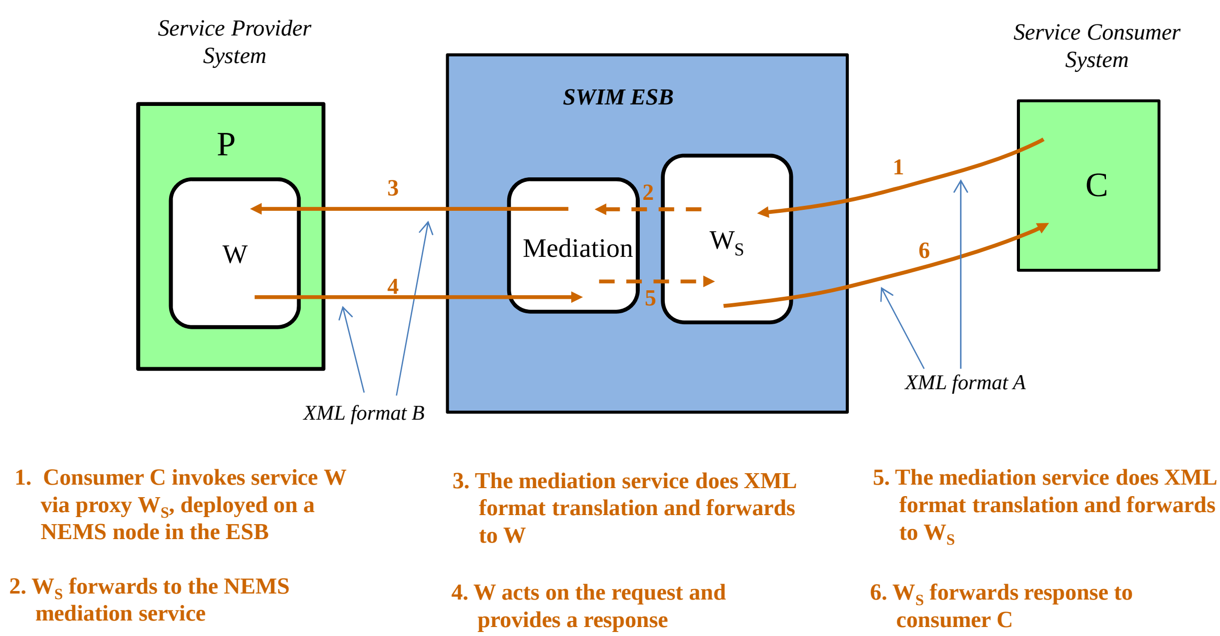


Figure 4-17. Mediation between Different XML formats

4.4.2 Support for NAS Legacy

Support for mediation to and from legacy message formats and protocol could be negotiated on a case-by-case basis during the on-ramping process. That is, there are no plans for specific transformations of NAS legacy message formats (e.g., Common Message Set (CMS)). The E1 Mediation service provides generic data transformation capabilities such as translation between different XML formats. Figure 4-18 shows how the NEMS mediation service could be used to transform NAS legacy message formats to XML as well as forward the same legacy formatted message that gets on-ramped by the producer to those consumers interested in the legacy format without any transformation.

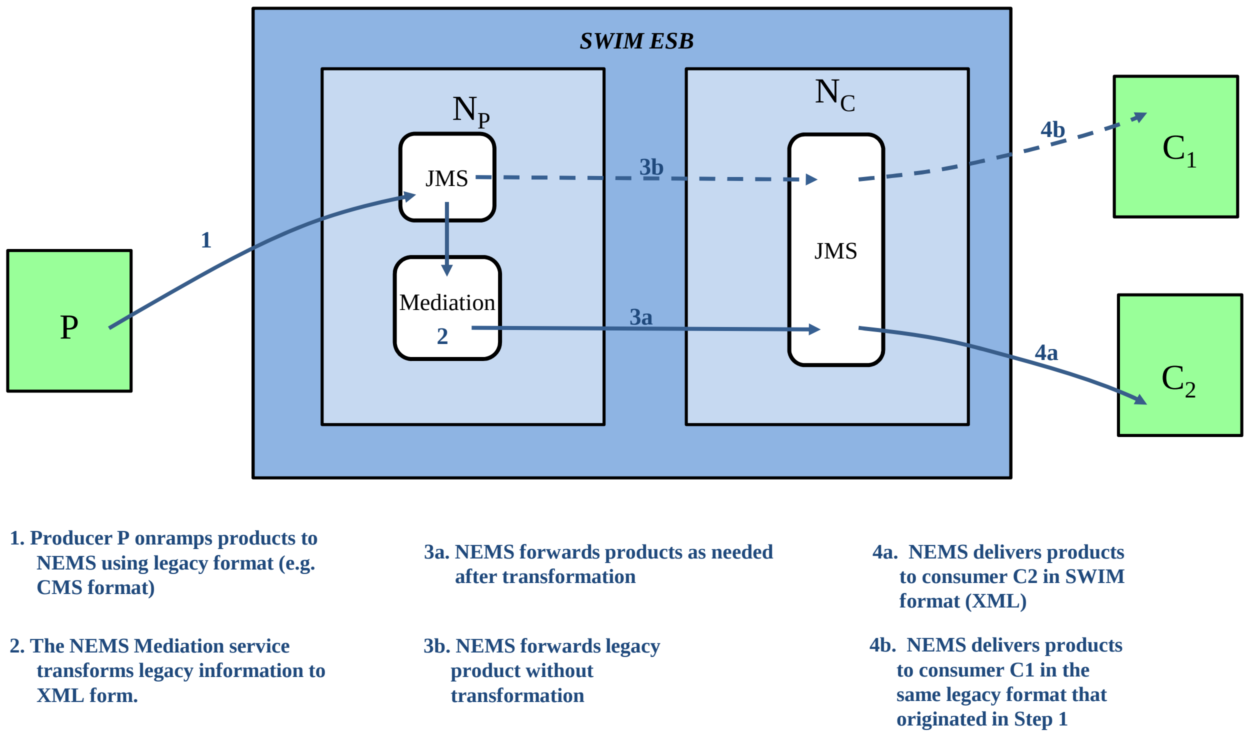


Figure 4-18. Mediation to and from Legacy (Non-XML) Formats

4.4.3 Support for Legacy COTS Business Applications

JCA adaptors provide the ability to integrate legacy COTS business systems with application servers. The JCA adaptor is either provided by the producer/consumer or can be developed by the NEMS service provider. In order to interoperate with NEMS using JCA, the Interoperability Service (E6) would have to be ordered.

4.5 Service Monitoring

This section covers two topics. The first topic is real-time monitoring of systems during normal operations in order to ensure that they are functioning correctly and, if not, to allow troubleshooting and corrective action to be initiated. We refer to this as operational monitoring. The second topic is monitoring in order to collect information so that the performance of systems can be determined and reported periodically. User programs may need these reports to ensure that various performance Service Level Agreements (SLAs) are being met. We refer to this as SLA monitoring.

4.5.1 NEMS Operational Monitoring and Maintenance

The FAA draft document *NEMS operational environment* [15] provides a description of the NEMS operational monitoring environment that is abstracted here. NEMS services require multiple pieces of infrastructure working together to produce each NEMS product flow. The SWIM Management Function (SMF) within the NEMC will monitor, manage, and provide notification and initial coordination of information Exchange Service interruptions or degradations. The FTI NOCC monitors and manages the Ops IP network and NEMS. The FTI SOC monitors and manages the Ops IP network security, including the NESG, and monitors and manages NEMS security.

Maintenance of the infrastructure needed to provide NEMS product flows is also performed by multiple organizations. In general, the System Support Center (SSC) maintains the FAA-owned producer and consumer end systems. Producers or consumers external to the FAA maintain their own equipment. The NEMS service provider (Harris Corporation) maintains the NEMS equipment that makes up the SWIM ESB, the equipment that makes up the Ops IP network, and the equipment used in the NESG.

Figure 4-19 illustrates the details of the SWIM Service Framework. User systems are connected via the POS and the COS to NEMS via the Ops IP network. While end-to-end monitoring is not available today, portions of the service strings are monitored by the FTI NOCC, and this information is made available to personnel at the NEMC.

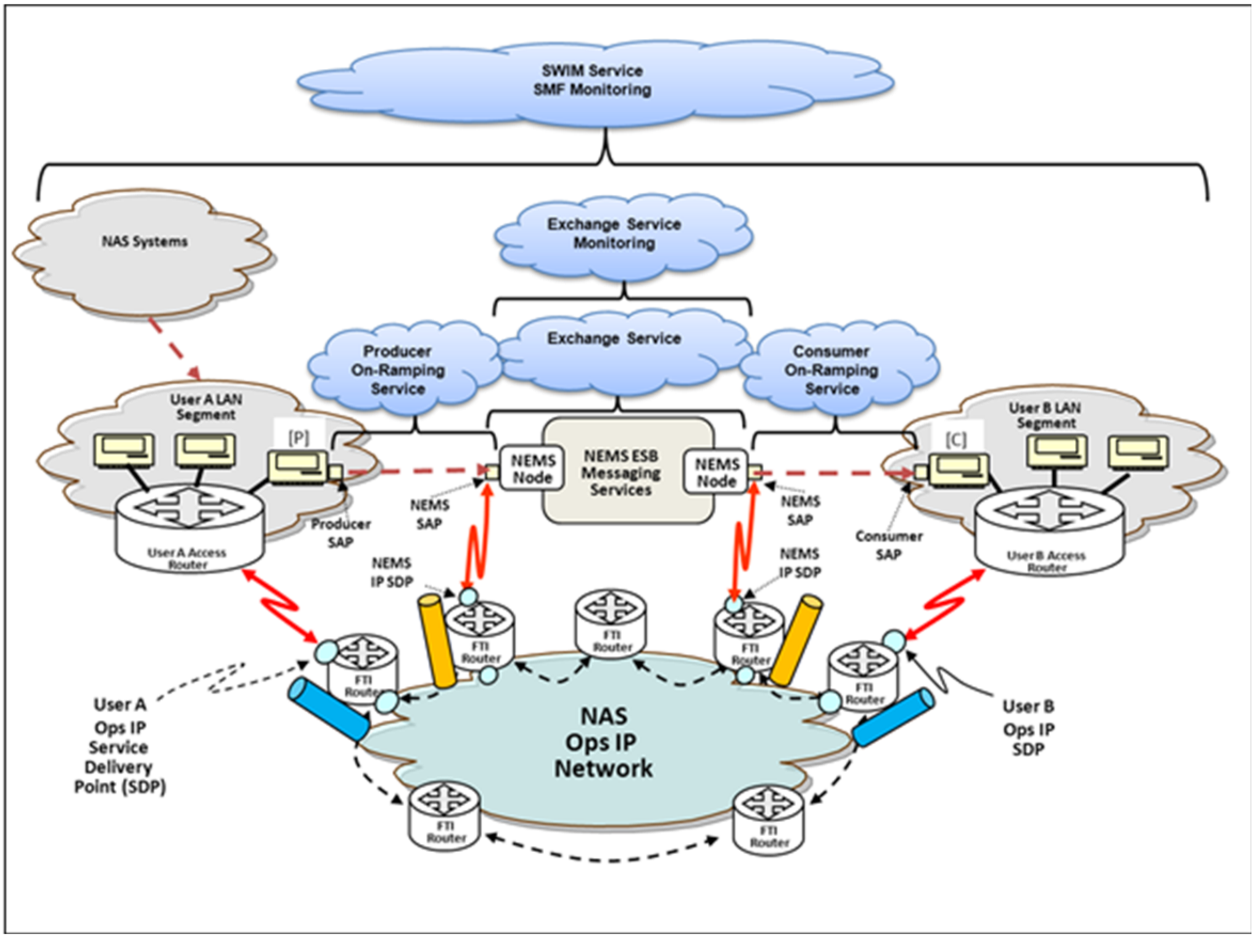


Figure 4-19. Detailed SWIM Service Framework

The NEMC is planned to be the single point of contact for producers and consumers. The SWIM Management Function (SMF) within the NEMC collaborates with personnel at the FTI NOCC, FTI SOC, and operators of producer and consumer systems. It monitors (via the help of the FTI NOCC) the up or down condition of the Producer SAP to Consumer SAP data flow. This includes the up or down condition of the: Producer SAP, Producer Service Definition Point (SDP), NEMS Producer SDP, NEMS Producer SAP, NEMS Infrastructure SDP, NEMS Consumer SAP, NEMS Consumer SDP, Consumer SDP, & Consumer SAP for every data flow. It reports on any scheduled or major unscheduled outages to producers and consumers and coordinates via the NOCC/SOC for all systems, services, and infrastructure that may affect NEMS end-to-end service flows. It reports on producer to consumer service SLAs and on the health of NEMS product flows.

Operators of producer/consumer systems are expected to report to the NEMC any interruptions or anomalies detected in the data being provided to NEMS, and any scheduled/unscheduled outages. They will also work with the NEMC, FTI NOCC, and consumers/producers in troubleshooting NEMS service outages and anomalies.

The NOCC and SOC will collaborate with NEMC, producers, and consumers to coordinate scheduled/unscheduled outages with the NEMC for all systems, services, and infrastructure that may affect NEMS end-to-end service flows. They will also coordinate NEMS, IP, and NESG outages scheduled/unscheduled with SWIM users.

The SSC collaborates with the NEMC, FTI Operations Control Center (OCC)/SOC to coordinate scheduled/unscheduled outages with OCC/SOC for NAS Systems producing or consuming data to or from NEMS.

It assists the NEMC, FTI Primary Network Operations and Control Center (PNOCC), Producers, and Consumers in troubleshooting NEMS Service outages and anomalies.

The NEMS nodes comprising the SWIM ESB will be monitored during regular operation. As shown in Figure 4-20, the scope of monitoring activities of the NEMS nodes is demarcated by the NEMS SAP. Three distinct segments of information flow are required for delivery of a SWIM product/sub-products from producer to consumer:

1. Producer to NEMS—Product for a NAS Business Service (Web Service or JMS message) flows into the SWIM ESB via a POS over the NAS Ops IP Network.
2. NEMS to NEMS—the ESB performs internal data/message transfers from node to node based on demand and message routing via the NAS Ops IP Network.
3. NEMS to Consumer—Data/message flows from the ESB to consumers (WS or JMS message) via a COS over the NAS Ops IP Network.

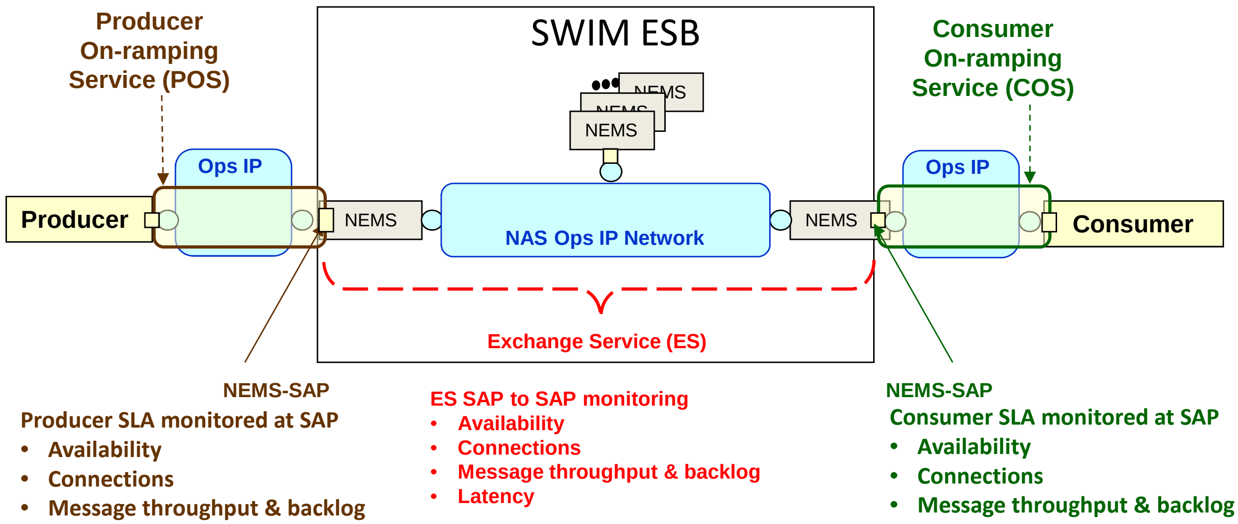


Figure 4-20. Description of NEMS Operational Monitoring

Within this frame of reference, it is expected that SLA levels, message throughput and backlog, availability and current levels of connections being used will be monitored.

These activities may be conducted on an ongoing real-time basis. Metrics collected from the SWIM ESB will be sent to the NOCC. Note that the NOCC will also be obtaining monitoring metrics pertaining to other systems (e.g., NAS Ops IP WAN). It is expected that this information obtained from the SWIM ESB will be used in conjunction with data from other systems for the network administrators and system engineers at the NOCC to maintain, troubleshoot, and identify faults and congestion issues. Currently, an automatic merging of data from these various systems is not possible and is coordinated with the NEMC personnel upon demand; the SWIM program plans to automate these exchanges.

4.5.2 NEMS SLA Monitoring and Reporting

The SLAs Service feature provides monitoring, reporting and management services to enable SLA monitoring. The features to monitor SLAs between producers and the NEMS or consumers and the NEMS are provided. The feature to monitor SLAs between producer and consumer is also provided. Standard SLA performance reports are provided on a regular basis (e.g., monthly). Subscription services for notifications and alerts to producers and consumers are also provided using existing NEMS JMS subscription service mechanisms. In order to obtain SLA reports, user programs should order the enhanced service “E4 SLAs” during on-ramping.

Users can also optionally order the enhanced service “E7 Reports” which provides the ability to generate custom reports providing summary and detailed performance metrics beyond the information provided by standard reports.

4.6 Orchestration

A user program can use orchestration services to create new functionality by coordinating other business services and maintaining state information on the enterprise service bus over a span of time.

The potential use of orchestration by a user program is best illustrated by an example. (Since user programs have not yet explored the possibilities of orchestration, this example is totally notional.) Imagine that user programs have already created and on-ramped the following business services:

- A “RequestClearanceForAirborneReroute” service that allows a NAS system to request that the Air Traffic Control (ATC) facility in control of a particular flight accept, and provide clearance for, a new route for that flight.
- A “CheckRouteAgainstRestrictions” service that will check whether a given flight route will intersect any restricted airspace.
- An “UpdateFlightStatus” service that will publish the current state of a given flight to all interested subscribers.
- A “RequestFlightRouteAirlineApproval” service that will determine whether a proposed flight routing is acceptable to the flight operators at an Airline Operations Center (AOC).

Given these services, a user program could then create a new service to be used for air traffic management: “RerouteFlight,” that functions as follows:

- First, invoke the CheckRouteAgainstRestrictions service for the flight. If it indicates that the route would traverse restricted airspace, reject the request, otherwise continue.
- Invoke the “RequestFlightRouteAirlineApproval service for the flight. If the service result indicates that the route is unacceptable to AOC, reject the request, otherwise continue.
- Invoke the RequestClearanceForAirborneReroute to request ATC approval to reroute the flight. If the request is accepted, invoke the “UpdateFlightStatus” service to notify all interested parties that the flight has been rerouted. Otherwise, send a response back to the entity that invoked the “RerouteFlight” indicating that the rerouting request has been rejected.

The above logic can be expressed in BPEL. The orchestration service allows a user program to provide business logic, expressed in BPEL, to be executed on the SWIM ESB to create new services that are built up from other business services, as in the example.

The orchestration service supports chaining together service invocations that may run over a long duration (e.g., minutes, hours, or days). For example, the “RequestFlightRouteAirlineApproval” service described above may need to wait for a human AOC operator to review and approve or reject the request. The orchestration service will maintain state information and support chaining together processes that run over a long duration.

When considering creating new functionality, all user programs are urged to review the existing business services listed in the NSRR, and consider whether these services can be orchestrated to create the necessary new functionality. If so, the SWIM orchestration service can be used rather than creating a new end system.

The orchestration service is available to user programs by ordering the “E3 Service Orchestration” service.

4.7 Publications to Research and Development

The R&D domain is a part of the SWIM infrastructure and provides a protected prototyping environment for development and testing activities for FAA & NextGen sponsored R&D activities; it also provides access to SWIM provided data and limited sets of legacy data. The

R&D domain comprises trusted and untrusted portions so as to accommodate both NAS and external users. Subscribing to this capability allows potential producers and consumers to publish and consume existing and potential products in this environment. SWIM normally expects that potential producers and consumers will be on-ramped to the R&D domain in the early phase of development. Development activities occur in the R&D domain with transition to the FNTB for inter-operability testing activities prior to final deployment to NAS.

5 List of References

[1] "FAA SWIM Website," Federal Aviation Administration, 31 March 2013. [Online]. Available: www.faa.gov/nextgen/swim. [Accessed 31 Mar 2013].

[2] J. Robb, "SWIM and NEMS Familiarization Package v12," Federal Aviation Administration, Washington, DC, Oct 2012.

[3] E. Zakrzewski, "NEMS Ordering Concepts and Terminology," Federal Aviation Administration, Washington, DC, Oct 2012.

[4] "SWIM Service Life Management Processes, v1.0," Federal Aviation Administration, Washington, DC, June 2010.

[5] "SWIM Service Life Management Processes, v2.0," Federal Aviation Administration, Washington, DC, Jun 2012.

[6] "NEMS Quarterly Program Review," The Harris Corporation, Melbourne, FL, Nov 2012.

[7] "SWIM Segment 2 Technical Overview, v1.2," Federal Aviation Administration, Washington, DC, Oct 2009.

[8] "FTI Operational IP User Guide, v3.3," Federal Aviation Administration, Washington, DC, Feb 2008.

[9] "AIXM - Aeronautical Information Exchange Model," FAA/Eurocontrol, March 2013. [Online]. Available: www.aixm.aero. [Accessed 31 Mar 2013].

[10] "SWIM Wiki," Federal Aviation Administration, Apr 2013. [Online]. Available: https://swimwiki.tc.faa.gov/.

[11] "SWIM KSN," Federal Aviation Administration, Apr 2013. [Online]. Available: https://ksn2.faa.gov/ajw/ajw-5/ajw-57/default.aspx.

[12] T. L. Bristol, *JO 1370.114: Implementation of FTI Services and Information Security Requirements in the National Airspace System (NAS),* Vice President, Technical Operations Services, effective Jan 4, 2012.

[13] S. Cooper, *JO 1370.116: Boundary Protection Policy,* Acting Deputy Assistant Administrator for Information Services and Deputy Chief Information Officer, effective April 16, 2012.

[14] "FTI Telecommunications Service Description, Appx J1, P050," Federal Aviation Administration, Washington, DC, 2012.

[15] R. Simmons, "NEMS Operational Environment v2-2," Federal Aviation Administration, Washington, DC, Jan 2013.

[16] "Java EE6 Tutorial," Oracle Corporation, [Online]. Available: http://docs.oracle.com/javaee/6/tutorial/doc/bncdr.html. [Accessed Apr 2013].

[17] "Java.net," Oracle, [Online]. Available: http://java.net/projects/jms-spec/pages/Home. [Accessed Apr 2013].

[18] "World Wide Web Consortium (W3C)," W3C, 1 Apr 2013. [Online]. Available: www.w3c.org. [Accessed 1 Apr 2013].

[19] "NEMS Categories of Capabilities and Services," Federal Aviation Administration, Washington, DC, Oct 2012.

[20] "NextGen Segment Implementation Plan 2010-2015, Version 4.0," Federal Aviation Administration, Washington DC, Apr 2012.

[21] E. Jalleta, B. Kaul and D. Thomson, "SWIM Core Architecture Evolution Concepts - Preliminary Draft," MITRE Corporation, McLean, VA, January 2009.

[22] "NextGen Implementation Plan," Federal Aviation Administration, Washington, DC, Mar 2012.

[23] "NextGen," Federal Aviation Administration, 2 Nov 2011. [Online]. Available: http://www.faa.gov/nextgen/.

[24] E. Jalleta, V. Prabhu and D. Thomson, "System Wide Information Management (SWIM) Evolution Considerations, MTR 120421," The MITRE Corporation, McLean, VA, Sept 2012.

Appendix A Introduction to JMS and Web Services

The two most prevalent messaging services for communicating between business applications are the Java Message Service (JMS) and the concept of a Web Service (WS). The following paragraphs provide an introduction (or a refresher) to JMS and WS for readers.

A.1 Java Messaging Service (JMS)

JMS is an Application Program Interface (API) for accessing enterprise messaging systems. It is part of the Java 2 Platform Enterprise Edition (J2EE). A wide range of enterprise messaging products such as IBM MQ support JMS so as to support reliable transport.

JMS is only an API standard; it does not provide an "on the wire" standard for machine-to-machine message transfer. Thus, implementations from multiple vendors may not interoperate even if they conform to the JMS API. In such cases, the enterprise either standardizes on a particular implementation or provides compatibility via its ESB.

The use of a Java language standard API is not intended to imply that SIPs must implement functionality in Java. Implementers using other languages may access the underlying messaging systems using other language APIs that may be provided as part of the particular messaging client software package in use; however, this should be done in such a way that other implementers can still access the messages using the JMS API.

Detailed tutorial material on JMS is available at the *Java EE6 Tutorial* website [16]; the proposed JMS 2.0 specification is available at *Java.Net* [17].

A.2 Web Services

Web services allow applications (possibly on a variety of computers using different operating systems) to share business logic, data and processes through a programmatic interface across a network. The applications interface with each other, not with the users. Web services are built on several technologies that work in conjunction with emerging standards to ensure security and manageability, and to make certain that Web services can be combined to work independent of a vendor. The term Web service describes a standardized way of integrating Web-based applications using the Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Web Services Description language (WSDL), and Universal Description, Discovery and Integration (UDDI) open standards over an Internet protocol network.

XML is used to tag the data, SOAP is used to transfer the data, WSDL is used for describing the services available, and UDDI is used for listing what services are available. Used primarily as a means for businesses to communicate with each other and with clients, Web services allow organizations to communicate data without intimate knowledge of each other's IT systems behind the firewall.

The World Wide Web Consortium (W3C) website [18] contains information on these standards.

Appendix B SWIM Service Lifecycle Management

Services that are made available to the NAS via SWIM are enterprise services and are subject to strict governance rules that are applicable throughout their lifecycle. This subsection is meant to introduce the topic, especially for new producers and consumers.

B.1 Producer SLM Stages

The *SWIM Service Lifecycle Management Processes* [5] defines the 7 stages and the required processes for service approval from conception to retirement. It also defines a pre-lifecycle management to describe the process that takes place before a service can be in the proposed stage. The process is illustrated in Figure B–1.

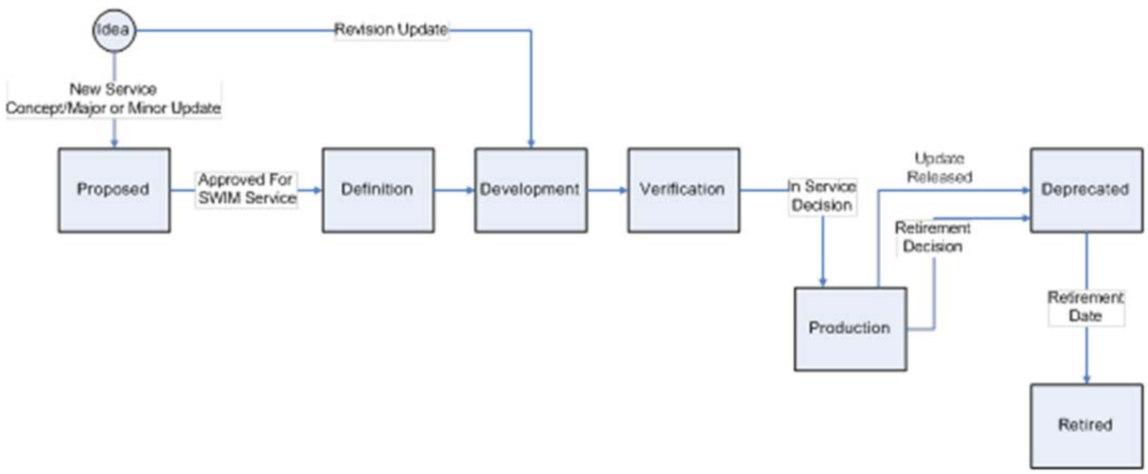


Figure B–1. SWIM Service Lifecycle Management Stages for Producers

The following 7 stages are defined for a service producer:

1. Service Proposal

   The main purpose of this stage is to identify what services will be provided by the Program via SWIM and to register them in the NSRR. The purpose of registering the service at this stage is to provide visibility to other programs as early as possible to avoid duplicate efforts. Early registration also will provide potential consumers an insight into future capabilities that they may not be aware of otherwise. It will give them an opportunity to participate in requirements development to shape service requirements to best satisfy their needs.

2. Service Definition

   The purpose of the Service Definition stage is to define, register, and approve the service contract. This stage is executed after the Service Proposed stage has been approved and a Final Investment Decision has been achieved.

3. Service Development

   The purpose of the Service Development stage is to develop and program the software to fulfill the Service Contract and implement the service. This stage is conducted after the Service Definition stage has been concluded.

4. Service Verification

   The purpose of the Service Verification Stage is to ensure that the design solution has met the system requirements and that the system is ready for use in the operational environment. Even though some of Development Testing and Verification activities occur during the Service Development stage, this stage will be conducted after the Service Development stage is concluded.

5. Service Production

   The purpose of the Production Stage is to monitor and ensure that the service operates according to the quality of service parameters set up by service level objectives, and according to FAA security policies.

6. Service Deprecation

   The purpose of the Deprecation stage is for operational services to be incrementally removed, either due to the development of new versions, or services that are no longer needed. A Deprecated service is an operational service that is scheduled to be retired based on the set retirement date. The service will be advanced to the Deprecation stage in the NSRR to caution current consumers of the intent to discontinue that service, and no new consumers will be allowed to consume the service.

7. Service Retirement

   The purpose of Service Retirement stage is to manage the removal of a service as per the governance process.

Starting from the Service Definition phase, a producer needs to consider SWIM capabilities and potential NEMS features of interest.

B.1.1 Pre-lifecycle Management

Ideas for proposed information Exchange Services may come from individual programs, Communities of Interest (COI), the NAS Enterprise Architecture (EA), and aviation user groups. In the NAS EA, these ideas may be in support of the NextGen Operational Improvements that are being considered. During SWIM Segment 1, potential services were proposed and decided by ATM domain-centric COIs (e.g., Flow-flight, Weather, and Aeronautical). Standards for information exchange for services are subject-specific standards that have been defined by the FAA and international stakeholders in the aeronautical community. Examples of these information exchange standards are Flight Information Exchange Model (FIXM), Weather Information Exchange Model (WXXM), and Aeronautical Information Exchange Model (AIXM). Information on these standards and their usage is available at the *AIXM website* [9]; the AIXM website provides links to the FIXM and WXXM websites.

It is assumed that any interested producers and consumers will use these standards or a combination of these standards.

B.1.2 Service Approval

Within the context of the SLM process “service approval” refers to the initial authorization to provide a SWIM-compliant service and subsequent approval of service artifacts and lifecycle advancement (e.g., from a Development stage to a Production stage, etc.). Service approval is contingent upon SWIM compliance. The SLM process describes the activities that should occur to confirm compliance with SWIM policies within each stage of the SLM process.

B.1.3 Service Registration

SWIM and FAA policies require that all SWIM-compliant services be registered in the NSRR and that the meta-information describing the service be provided. Registration of service meta-information (meta-data and service artifacts) occurs throughout the SLM process, and must be submitted to the NSRR at the required lifecycle stage. Registration of a service and submission of associated meta-information within the NSRR promotes early discovery and information sharing. All services in the NSRR are assigned a lifecycle stage. The SWIM Lifecycle Taxonomy defines an ordered set of values for the 7 service stages as shown in Table B–1.

Table B–1. NSRR Lifecycle Taxonomy for SLM Stages

Name	Description
Proposed	Indicates that the service has been proposed as a SWIM compliant service.
Definition	The service proposal has been approved allowing the provider to proceed with definition of the service contract.
Development	The service contract has been approved allowing the provider to proceed to service implementation. The service is under development; it may not be placed under version control and could be changed without notification.
Verification	The service is ready for system-level testing based on the outcome of a Test Readiness Review.
Production	The registration authority has confirmed that the service is of sufficient quality and is compliant with the set of standards and regulations in the community that uses this service registry, and is therefore approved for use.
Deprecated	The service is scheduled for retirement and may not accept new consumers.
Retired	The service may no longer be used.

B.1.4 Service Definition

SWIM endorses a “Contract First” approach for service development. This means that the service contract is developed and defined prior to code implementation. It is in congruence with the current Acquisition Management System practice of defining an Interface Requirements Document (IRD)/Web Service Requirements Document (WSRD) and an Interface Control Document (ICD)/ Web Service Description Document (WSDD). This approach does not require a final approved WSRD and WSDD prior to submission of the service contract; however, the WSRD and WSDD should be far enough along in their development to minimize subsequent updates to the service contract due to changes.

A service contract is composed of one or more published documents that express meta-information about a service.

A *Service Contract* defines:

- *What* the service’s purpose and capabilities are.
- *How* the service can be accessed (i.e., communication and transport protocols).
- *Where* the service can be accessed (i.e., its location).
- The *terms of engagement* (i.e., the service level agreement between the provider and consumer).

Within the context of SWIM, a service contract is defined as a composition of:

- *Technical Service Contract*. The technical service contract is composed of a WSDL, associated XML schemas for type definitions, and, when applicable, WS-Policy assertions[15]. SWIM SOA Governance policy requires every service (not just SOAP-based Web services) to have a WSDL if practical.
- One or more *Service Level Objectives* (SLO). An SLO is a document that defines a named set of Quality of Service (QoS) parameters and level of commitment of the organization/service owner to meet the expectations detailed within it. A list of suggested QoS parameters describing anticipated service behavior includes:
  - Level of service availability.
  - Level of service capacity.
  - Service performance measurements, like throughput and response time.

Once the service is in Production, a potential consumer will select the service based on the QoS parameters that are defined within the SLO. When consumption is approved, the SLO becomes a de facto Service Level Agreement (SLA).

B.2 Consumer SLM Stages

The Consumer Service Lifecycle Management Process describes the governance processes for service discovery and consumption. A high-level overview of this process is depicted in Figure B–2.

The NSRR provides a centralized mechanism for discovery of SWIM-based services. The NSRR is the “system of record” for SWIM-based services. Discovery of SWIM-based services and their capabilities should occur via the NSRR and not via other mechanisms that could lead to incompatibilities. To gain access to the NSRR for service discovery, interested parties must request authorization and follow the process for service discovery by authorized NSRR users.

[15] Typically in the FAA technical products such as the WSDLs and XML schemas that define the service contract are considered source code, and are tested and approved through Line of Business level configuration management processes. However, because of the importance of service contracts in an SOA environment for visibility of services and technical interoperability, approval of service contracts has been moved up earlier in the development cycle. Once approved and placed under configuration control, changes to the service contract will be submitted to the NSRR and approved following the processes described in the Service Definition Stage of the SLM Process.

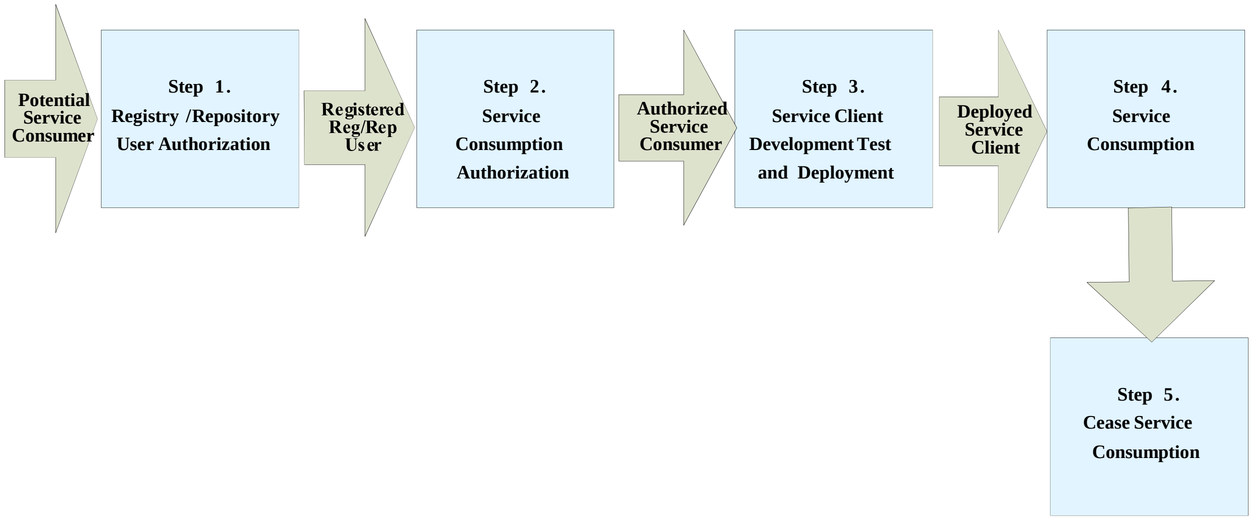


Figure B–2. SWIM Service Life Management for Consumers

Once an interested party has identified a SWIM-based service they would like to consume, they must request authorization to do so. Service Consumers must consume services in accordance with a service contract; there are no anonymous consumers. Service Consumers must negotiate the SLA component of the service contract with the Service Producer.

Authorized Service Consumers will use the meta-information in the NSRR to develop their client software. SWIM Policy requires that Service Producers make available a Reference Consumer Implementation to aid consumers in their development of service clients. Service Producers are responsible for defining and enforcing requirements for testing service clients. Prior to deploying client software, Service Consumers must use an emulated service for integration testing if one is offered by the Service Producer.

Once a client is deployed, SWIM policy requires that Service Consumers manage their service consumption to ensure that their usage is within agreed upon parameters. Service Consumers have a shared responsibility with Service Producers to monitor the service for compliance with the QoS levels specified in the SLA.

Service Consumers must also remain aware of the service lifecycle stage of consumed services. The NSRR will notify Service Consumers when services are deprecated. Consumers of Deprecated stage services must migrate to the new service version or an alternative solution prior to the retirement date established for the deprecated service.

Appendix C NEMS System and On-Ramping Services

The NAS Enterprise Messaging Service (NEMS) capabilities are being acquired from the Federal Telecommunications Infrastructure (FTI) service provider and have been classified as on-ramping services and system level capabilities. This description has been extracted from the FAA draft document *NEMS Categories of Capabilities and Services* [19]. A summary of these categories is shown in Figure C–1. The capabilities are described below.

NEMS system level capabilities are those that the System Wide Information Management (SWIM) program can order for NEMS as a whole. These include the basic capabilities (B1-B6, B12); in addition, optional capabilities (B7-B11, B13, B14) can be ordered. Once ordered and incorporated into NEMS these capabilities are available for the use of all producers and consumers. On-ramping services are services/features that are pertinent to a producer or a consumer to connect to the SWIM Enterprise Service Bus (ESB); a producer or a consumer has to order one or more of the basic on-ramping features (U1-U4) and may choose to order one or more of the enhanced features (E1-E9).

System Level Capabilities

Basic
- **B1: Run-time Subscription**
- **B2: Message Reliability QoS**
- **B3: Mediation**
- **B4: Availability and Performance**
- **B5: Security Services**
- **B6: Web Services**
- **B12: Dynamic Subscriptions**

Optional
- **B7: Run-time Registry**
- **B8: Interoperability**
- **B9: Service Orchestration**
- **B10: Producer/Consumer SLAs**
- **B11: Enterprise Repository**
- **B13: Global Load Balancers**
- **B14: Local Load Balancers**

On-Ramping Services/Features

Basic
- **U1: WS-C: Web Service - Consumer**
- **U2: WS-P: Web Service – Producer**
- **U3: JMS-C: JMS Subscription - Consumer**
- **U4: JMS-P: JMS Publishing- Producer**

Enhanced
- **E1: Mediation**
- **E2: Performance**
- **E3: Service Orchestration**
- **E4: SLAs**
- **E5: Security**
- **E6: Interoperability**
- **E7: Reports**
- **E8: Dynamic Subscriptions**
- **E9: Publications to R&D**

Figure C–1. NEMS System Capabilities and On-Ramping Services

On-ramping enhanced services/features obviously depend on the system level capabilities that are available in the NEMS. For example, a user cannot order the E3 Service Orchestration feature unless the SWIM program had already ordered the optional B9 service either by itself or for another user program. In the event that a user requires an on-ramping feature that is not yet supported on the NEMS, then that specific on-ramping feature as well as the optional system level capability will have be ordered together. The dependency of specific on-ramping features on system level features is discussed in C.3.

C.1 System Level Capabilities

C.1.1 Basic System Level Capabilities

The NEMS baseline services provide the following capabilities:

- **B1 Run-time Subscription Configuration**

  The Run-time Subscription Configuration capability provides a NEMS hosted web service that provides the ability to create, read, update, and delete Java Messaging Service (JMS) product subscription information stored in the NEMS content catalog (registry). This interface will be made available to authorized consumers per FAA defined governance policy. Access to specific content within the content catalog will be controlled to ensure a consumer can only access authorized content information and ensure a consumer cannot over subscribe to content leading to an unacceptable utilization of network and/or NEMS processing resources. This control will be implemented by developing role-based access control on a content catalog entry basis. As part of the on-ramping process for consumers, the content a consumer can subscribe to will be determined and configured for subsequent enforcement during run-time access.

- **B2 Messaging Reliability QoS**

  The Messaging Reliability Quality of Service (QoS) capability provides guaranteed message delivery for JMS and web service (WS) users. Guaranteed delivery ensures that once a message is received at a NEMS Service Access Point (SAP), it is delivered to the SAP of the consumer or consumers once and only once. A service access point is a queue or a topic on a NEMS node for JMS services and web service proxy on a NEMS node for web services. This capability also provides durable subscriptions for JMS consumers. Durable subscriptions allow for intentional or unintentional disconnects by consumers to the NEMS JMS service without losing subscribed content published during the disconnection period. Upon reconnection the missed subscribed content is delivered to the consumer. The disconnection interval will be configurable per consumer topic. In order to support both of these messaging QoS, secondary data storage is necessary to persist messages while in transit and will be deployed to the NEMS nodes providing these services. In order to support business continuity in the event of a site outage, in transit messages will be replicated to the designated backup site.

- **B3 Mediation**

  The Mediation capability provides message content and protocol transformation capabilities for NEMS producers and consumers. Message content transformation consists of reformatting or changing message payload content, as it transits through the ESB, per a set of configurable rules. Extensible Stylesheet Language Transformations (XSLT) and XQuery are used to specify and perform transformations on textual message payloads. All message content transformations can be configured on a consumer destination basis for JMS consumers.

  Protocol transformation enables protocol disparate producers and consumers to communicate without having to modify either producer or consumer software. The NEMS Oracle Service Bus (OSB) is utilized to provide protocol transformations for JMS and WS producers and consumers. Interoperability between WebLogic and ActiveMQ is also supported.

- **B4 Availability and Performance Monitoring**

  The Availability and Performance Monitoring capability provides monitoring, reporting and management services to enable RMA3 and RMA4 NEMS service availability including backup site designation and configuration in order to support business continuity. Enhanced service management services provide the ability to monitor, record and alert based on various system health and performance parameters sufficient to establish an audit trail and support event analysis. The capability to monitor SLAs between producers SAP and the NEMS, as well as that between consumers SAP and the NEMS is provided. The ability to generate standard and custom reports providing summary and detailed metrics is provided. Subscription services for notifications and alerts to producers and consumers are provided using existing NEMS JMS subscription service mechanisms.

- **B5 Security Services**

  The Security Services capability provides enhanced authentication and service level access control including integrity, privacy, and encryption for NEMS services specifically WS and JMS. JMS services are secured using Transport Layer Security (TLS) providing secure session level encryption. Web services are secured in accordance with SWIM Basic Security Profiles, SWIM Web Service Security Profiles and WS security standards.

  Role based access control is provided utilizing Oracle Entitlements Server (OES) for WS and JMS access by producers and consumers. Authentication control is provided for all producers and consumers utilizing Oracle Internet Directory (OID). WebLogic and ServiceMix provide the policy enforcement points to enforce the policies created using OID and OES. Access control for JMS producers and consumers is provided at the queue/topic and operations level ensuring only authorized users can read/write to a specific topic/queue. Authentication control for web services supports multiple credential packaging mechanisms including X.509 digital certificates, username tokens, binary username tokens and Security Assertion Markup Language (SAML) tokens. The Security Service capability provides the ability to collect and log security events to support audits and provides report generation tools for analysis of security event data.

  OSB is utilized to perform extensible markup language (XML) schema validation for selected message payloads and blocking message delivery of messages failing schema validation.

- **B6 Web Services**

  The Web Services capability provides support for request/response messaging utilizing Simple Object Access Protocol *(*SOAP) and Restful web services. WS standards for addressing, reliable messaging, and notification are supported. The NEMS web service implementation is compliant with SWIM WS-Interoperability policies. Web service interaction between the producer and consumer is facilitated using web service proxies utilizing http/https transport protocols. The proxies provide the ability to support service level access control and provide a centralized point for monitoring all web service performance and availability. SOAP web services provide the capability to support SOAP with attachments and Message Transmission Optimization Mechanism (MTOM) mechanisms.

- **B12 Dynamic Subscription**

  The Dynamic Subscription capability provides the capability to support the initiation of a subscription by a consumer for consumer specified content provided by a producer via the NEMS. The initiation of the subscription by the consumer is accomplished by invoking a web service proxy hosted on the NEMS. The web service proxy acts as an intermediary to set up a subscription with the producer and returns a NEMS resident JMS topic or queue for the consumer to subscribe to using the ActiveMQ JMS Application Programmer Interface (API). The service can optionally return a uniform resource locator (URL) to a file.

C.1.2 Optional System Level Capabilities

The NEMS baseline services described previously can be enhanced to provide the following System Level Enhanced capabilities as ordered by the FAA.

- **B7 Run-Time Registry**

  The Run-time Registry capability provides a *Universal Description, Discovery and Integration* (UDDI) run-time service registry for administratively controlled web service registration for runtime access by clients. The service registry is used for web service endpoint lookup by clients during runtime. This approach enables web services to be deployed to alternate locations without impacting the client. The OSB is the client of the service registry as the service registry is transparent to NEMS consumers. NEMS consumers simply access the applicable web service proxy and the NEMS performs the function of determining the business service endpoints utilizing the service registry.

- **B8 Interoperability**

  The Interoperability capability provides the capability for the NEMS to interoperate with a Java Connector Architecture (JCA) adaptor. JCA adaptors provide the ability to integrate legacy systems with application servers and leverage the robust capabilities of the application server. WebLogic is a JCA compliant application server and supports JCA adaptors. An example JCA adapter will be configured to interoperate with WebLogic and a demonstration will be provided.

- **B9 Service Orchestration**

  The Service Orchestration capability provides a Business Process Execution Language (BPEL) engine to support BPEL driven service orchestration of web services. The Oracle BPEL Process Manager is utilized to facilitate development of SOA applications through composing synchronous and asynchronous services into end-to-end, standard BPEL process flows.

  The BPEL Process Manager includes a BPEL Designer that provides a graphical and user-friendly way to build BPEL processes. The BPEL Designer uses BPEL as its native format so that processes built with the Designer are portable. The BPEL engine leverages an underlying J2EE application server, with support for most major commercial application servers. The built-in integration services enable developers to easily leverage advanced connectivity and transformation capabilities from standard BPEL processes. These capabilities include support for XSLT and XQuery transformation as well as bindings to legacy systems through JCA adapters and native protocols. Human workflow is provided as a built-in BPEL service to enable the integration of people and manual tasks into BPEL flows.

The BPEL Process Manager is deployed on redundant dedicated servers connected to the operational network at a FAA selected location. Rudimentary sample applications will be developed to demonstrate BPEL Process Manager capabilities.

- **B10 Producer/Consumer SLA**

  The Producer/Consumer SLA capability provides the functionality to establish and monitor SLAs between producers and consumers. Oracle Enterprise Manager (OEM) will be utilized to monitor SAPs for producers and consumers and all NEMS services at each node participating in the service delivery. Reports are provided that provide summary and detailed metrics associated with the SLA.

- **B11 Enterprise Repository**

  The Enterprise Repository capability supports design-time SOA governance for NEMS. Oracle Enterprise Repository (OER) is utilized to provide flexible and comprehensive end-to-end service lifecycle SOA governance features. OER provides the ability to create custom metadata models providing the flexibility to support the information specific to enterprise needs. OER also provides workflow management for automation of service portfolio management over the full lifecycle of a SOA service.

  The Enterprise Repository is deployed on redundant dedicated servers connected to the mission support network at a FAA selected location. A rudimentary sample workflow will be developed to demonstrate service lifecycle governance. The default metadata model provided with OER will be populated sufficiently to support the rudimentary sample workflow.

- **B13 Global Load Balancing/Failover**

  The Global Load Balancing and Failover capability provides Global Load Balancers deployed in the NAS. This capability enables NEMS processing loads to be balanced across multiple sites. This capability also provides for detecting site-wide failures and routing requests to alternate NEMS sites. Global Load Balancers are deployed to selected sites remote to the NEMS nodes, to provide service routing capabilities based on system health conditions of geographically distributed NEMS nodes.

- **B14 Local Load Balancing**

  The Local Load Balancing capability deploys Local Load Balancers to NEMS node sites to provide local load balancing capabilities based on the load and health conditions of the local NEMS servers. This capability provides for detecting server failures and routing requests to alternate local servers providing high availability of local NEMS messaging services. Local Load Balancers deployed at each NEMS site also provide performance optimization resulting from effective resource utilization of NEMS services.

C.2 On-Ramping Services

On-ramping Services offer an entry-level service for NEMS producers or consumers. Selection of at least one of these services is mandatory for on-ramping a producer or consumer to the NEMS. Additional services for advanced capabilities are described in the enhanced on-ramping capabilities section.

C.2.1 Basic On-Ramping Features

- **U1: WS-C Web Service Consumer**

  The Web Service Consumer service provides for the on-ramping of a consumer in order for the consumer to access web services being proxied by the NEMS. The consumer is provided access to the web service definition language (WSDL) instances as well as username token security credentials for accessing the required web services. The NEMS is configured to allow the consumer access to the hosted proxies for the required web services. Service monitoring performance thresholds are updated as necessary as a result of additional expected message traffic associated with the anticipated web service access. Standard reports are provided offering summary and detailed metrics regarding NEMS service performance. Best effort messaging is provided. Local and Global Load Balancers are configured, if ordered, as necessary to support the new consumer.

- **U2: WS-P Web Service Producer**

  The Web Service Producer service provides for the on-ramping of a producer in order to make web services available to authorized NEMS consumers. Each web service is proxied by the NEMS using a WSDL that mirrors the WSDL used by the producer web service with exception of the end-point. Service monitoring is configured in order to monitor health, status, and resource utilization of the proxy. Security access controls are configured for the proxy based on policies defined for access to the web service. Best effort messaging is provided. Standard reports are provided offering summary and detailed metrics regarding NEMS service performance. Local and Global Load Balancers are configured, if ordered, as necessary to support the new producer service.

- **U3: JMS-C Subscription Consumer**

  The Subscription Consumer service provides for the on-ramping of a consumer in order for the consumer to access subscription services being offered by the NEMS. The consumer is provided access to the read-only version of the Navigator to view available subscription content. Desired content is configured for routing to the consumer and username/password security credentials are provided for access to an assigned topic by the consumer application. Web Service access to the content catalog may be made available to the consumer in accordance with SWIM governance policy. Service monitoring is configured to monitor the assigned topic to determine that messages are being delivered. WebLogic and ActiveMQ JMS consumers are supported. Best effort messaging is provided. Standard reports are provided offering summary and detailed metrics regarding NEMS service performance. Local and Global Load Balancers are configured, if ordered, as necessary to support the new consumer.

- **U4: JMS-P Subscription Producer**

  The Subscription Producer service provides for the on-ramping of a producer for publishing content to the NEMS via JMS. A taxonomy of published content including subsets, available for subscription by consumers, is defined, implemented, and deployed to the content catalog. Username/password security credentials are provided for access to the assigned queue by the producer application. Service monitoring is configured to monitor the assigned queue to determine that messages are being delivered. WebLogic and ActiveMQ JMS producers are supported. Standard reports are provided offering summary and detailed metrics regarding NEMS service performance. Best effort messaging is provided.

C.2.2 Enhanced On-Ramping Capabilities

Enhanced on-ramping capabilities offer advanced-level capabilities for producers or consumers. Selection of these capabilities is optional for on-ramping a producer or consumer to the NEMS.

- **E1 Mediation**

  The Mediation capability provides message content and protocol transformation capabilities for NEMS producers and consumers. Required message content transformations are developed and deployed to reformat or change message payload content, as it transits through the ESB, per a set of configurable rules defined using XSLT and/or XQuery. The NEMS ESB is configured to provide the required protocol transformations for JMS and/or WS producers and consumers.

- **E2 Performance**

  The Performance capability provides guaranteed message delivery for JMS and WS consumers as well as durable subscriptions if required. JMS guaranteed delivery parameters are configured to ensure that once a message is received at a NEMS SAP, it is delivered to the SAP of the consumer or consumers once and only once. Durable subscriptions are configured, including the connection interval, to allow for intentional or unintentional disconnects by consumers to the NEMS without losing subscribed content published during the disconnection period. In order to support both of these messaging QoS, persistent data stores are necessary and must have been previously deployed to the participating NEMS nodes. Priority based messaging is also provided which allows a JMS publisher to indicate a priority in the message header. The highest priority message will be removed from a queue and processed before lower priority messages are processed. This capability supports both WebLogic and ActiveMQ JMS consumers.

- **E3 Service Orchestration**

  The Service Orchestration capability provides the development and deployment of BPEL driven service orchestration flows utilizing currently deployed services. The Oracle BPEL Process Manager is utilized to facilitate development of SOA applications through composing synchronous and asynchronous services into end-to-end, standard BPEL process flows. Human workflow is provided, if required, as a built-in BPEL service to enable the integration of people and manual tasks into BPEL flows.

- **E4 SLAs**

  The SLAs capability provides monitoring, reporting and management services to enable SLA monitoring. The capability to monitor SLAs between producers and the NEMS or consumers and the NEMS are provided. The capability to monitor SLAs between producer and consumer is also provided. Standard SLA performance reports are provided. Subscription services for notifications and alerts to producers and consumers are provided using existing NEMS JMS subscription service mechanisms.

- **E5 Security**

  The Security capability provides advanced service level access control including integrity, privacy, and encryption for NEMS services specifically WS and JMS.

Advanced capabilities include TLS session level protection using X.509 digital certificates for JMS services and binary username token, X.509 digital certificates, and SAML tokens for web services. OSB is utilized to perform any required XML schema validation for selected message payloads and blocking of message delivery of any messages failing schema validation.

- **E6 Interoperability**

  The Interoperability capability provides the capability for the NEMS to interoperate with a producer or consumer via a JCA adaptor. JCA adaptors provide the ability to integrate legacy systems with application servers and leverage the robust capabilities of the application server. WebLogic is a JCA compliant application server and supports JCA adaptors. The JCA adaptor is either provided by the producer/consumer or can be developed by Harris.

- **E7 Reports**

  The Report capability provides the capability to generate custom reports providing summary and detailed performance metrics beyond the information provided by standard reports for producers and consumers.

- **E8 Dynamic Subscription**

  The Dynamic Subscription capability provides the capability to support the initiation of a subscription by a consumer for consumer specified content provided by a producer via the NEMS. The initiation of the subscription by the consumer is accomplished by invoking a web service proxy hosted on the NEMS. The web service proxy acts as an intermediary to set up a subscription with the producer and returns a NEMS resident JMS topic or queue for the consumer to subscribe to using the ActiveMQ JMS API. The service can optionally return a URL to a file. The Dynamic Subscription capability can be used to broker OGC publish/subscribe services in addition to any other producer provided subscription services that support this messaging pattern.

- **E9 Publications to R&D**

  The Publications to R&D permits SIP producers and consumers access to a protected environment for development and testing activities for FAA and NextGen sponsored R&D activities. Access is provided to SWIM provided data and limited sets of legacy data. Development activities that occur in the R&D Domain are expected to transition to the FTI National Test Bed (FNTB) for inter-operability activities prior to final deployment to NAS.

C.3 Mapping of On-Ramping features to System Level capabilities

The enhanced on-ramping features that may be needed by programs are associated with certain system level capabilities and assume that the latter are already available (see Table C–1). If a system level capability is not already available, these will have to be ordered as a set along with the enhanced on-ramping feature.

Table C–1. Mapping of Enhanced On-Ramping Features to System Level Capability

Associated Enhanced On-Ramping Features	System Level Capability

Associated Enhanced On-Ramping Features	System Level Capability
E1: Mediation	B3: Mediation
E2: Performance	B2: Message Reliability QoS
E3: Service Orchestration	B9: Service Orchestration
E4: SLAs	B4: Availability and Performance and B10:Producer / Consumer SLAs
E5: Security	B5: Security Services
E6: Interoperability	B8: Interoperability
E7: Reporting	B4: Availability and Performance
E8: Dynamic Subscriptions	B12: Dynamic Subscriptions

Appendix D Glossary and Abbreviations

D.1 Glossary

Accessible	A service that may be consumed by means of either the request/response or publish-subscribe operational pattern is *accessible*.
Application developer	A person responsible for developing an application or system.
Application End System	Denotes both producer and consumer systems; also a system that performs both functions
Authorization	Permission to engage in a specific activity. A SWIM user is *authorized* if they have permission to engage in a specific activity, such as subscribing to a publication service.
Build-time	The lifecycle stage in which a service provider or consumer is under development, e.g., pre-operational. Also called *design-time*.
Capability	A grouping of high-level functions implemented as a service or a group of services that may contain applications that are not service oriented software implementations.
Community of interest (COI)	A collaborative group of users who must exchange information in pursuit of shared goals, interests, missions, or business processes. COIs are established in a variety of ways and may be composed of members from one or more functions and organizations as needed for a shared mission.
Consumer	An executable process, running on a server that accesses one or more SWIM business services and associated content via the SWIM ESB. (also NEMS Consumer or Service Consumer)
Consumer Program	Program organization that sponsors the on-ramping of a NEMS Consumer for the purposes of accessing content of one or more business services; these can be NAS or non-NAS programs
Consumer System	A system that connects to the SWIM ESB in order to consume service content; a consumer system will contain one or more consumers
Consumer On-Ramping Service (COS)	Application (layer 7) service access connection that enables an association between the NEMS consumer of a user program and the SWIM ESB to consume service content
Content Based Routing (CBR)	Routing based on the content of the message; in NEMS these fields are predefined and carried in the header

Core services	The fundamental SWIM mechanisms that enable consumption of services and information sharing: Interface Management, Messaging, Enterprise Service Management, and Security. These services are solution-agnostic (not limited to a single process or solution environment) and have a high degree of autonomy so that they support reuse.
Design-time	The lifecycle stage in which a service provider or consumer is under development, e.g., pre-operational. Also called *build-time*.
Developer	Person or persons responsible for developing a service provider or consumer.
Discoverable	A service that may be discovered by a potential user is *discoverable*.
Discovery	The act of locating and accessing the schema for a specific service.
Dissemination	The act of distributing data to one or more recipients.
Durable Subscription	Durable subscriptions enable consumers to recover subscribed content after a planned or unplanned disruption of service or disconnection while publication is occurring.
Enterprise Service Management (ESM)	The SWIM core service addressing the management of SWIM-based services, including performance and availability. ESM provides the ability to monitor, manage, and scale services within the enterprise to ensure the capability offerings are available, responsive, and scalable to the operational environment supported.
Exchange Service	SWIM ESB service management capability that brokers business service content (products and sub-products) between producers and consumers
Expose	To make a service interface discoverable. In SWIM, services are exposed via the SWIM NSRR.
Extensibility	A characteristic of an interface (or service) that continues to support previously conformant users after it has been modified (i.e., *extended*) for new users.
Filtering rules	Filtering rules define constraints on a service provider with respect to the data to be provided to a consumer.
Interface management	The SWIM core service providing a standard interoperable means for description, access, invocation and manipulation of resources to enable compatible communications between SWIM service producers and consumers.
Infrastructure	The logical and physical (i.e., hardware and software) elements that together provide (SWIM) functionality.
Java Messaging Service (JMS)	An application program interface for accessing enterprise messaging systems; part of the Java 2 Platform Enterprise Edition

Message	A structured information exchange package consisting of a header and payload
Messaging	The SWIM core service that provides delivery of data and notifications between applications and systems.
NAS Enterprise Messaging Service (NEMS)	An enterprise messaging service provided by SWIM
NAS Enterprise Security Gateways (NESG)	Gateway that provides NAS boundary protection services
NAS Service Registry Repository (NSRR)	A static registry or directory containing entries with the information necessary to discover and access services. The Registry utilizes a formal registration process to store, catalog, and manage metadata relevant to the services, thereby enabling the search, identification, and understanding of resources. Also referred to as "Service Registry" or "Registry."
NEMS Service Access Point (NEMS-SAP)	SWIM ESB egress point for either sending or receiving NEMS messages to/from the SWIM ESB
Notification	An indication presented to a user regarding the status of a system or an element in a system. In a publish-subscribe system, a publication may consist of notifications about data rather than the data itself.
On-Ramping	Application (layer 7) service access connection enabling an association between a producer/consumer and the SWIM ESB for the providing, consuming, or exchanging service content
Operational Pattern	An operational pattern describes the essential flow of a SWIM-based service. It is based on the term pattern, which describes the essential features of a common solution to a common problem in software development.
Producer	An executable process, running on a server that provides a SWIM business service and associated service content to the SWIM ESB for purpose of being accessed by consumers
Producer On-Ramping Service (POS)	Application (layer 7) service access connection enabling an association between a producer and the SWIM ESB for the purposes of providing service content
Producer Program	Program organization that sponsors the on-ramping of a NEMS producer and the SWIM business services provided by that producer; these can be NAS or non-NAS organizations
Producer System	A system that connects to the SWIM ESB and produces JMS messages or provide a web service; a producer system will contain one or more producers
Provider	Service provider.

Publication	A service based on the publish-subscribe operational pattern.
Publisher	A service provider utilizing the publish-subscribe operational pattern.
Publish-subscribe	A one-to-many operational pattern in which a service provider called a *publisher* makes its services available (i.e., publishes) on a subscription basis. A service consumer in this paradigm called a *subscriber* requests access to the publication service via a subscription request. Based on the nature of their subscriptions, subscribers will continue to receive updates from the publisher until they request the termination of their subscription.
Reconstitution	The process of bringing a subscriber "up to date" on previously published data.
Register	The act of exposing a service by placing its location and schema information in the SWIM NSRR.
Reliable delivery	A characteristic of information transfer in which the transfer is either successful or the sender of the information is notified of the failure of the transfer.
Request/response	The operational pattern distinguished by a two-way interaction between a requesting entity and a responding entity.
Runtime	The lifecycle stage in which a service provider or consumer is operational.
Security	The SWIM core service responsible for the protection of information, operation, assets, and participants from unauthorized access or attack.
Selection Criteria	Selection criteria provide the means by which a consumer identifies the data of interest to a service provider.
Service	An implementation-independent reusable operational function that may be discovered as self-describing interfaces, and invoked using open standard protocols across networks. Services can be combined and orchestrated to produce composite services and operations processes, in accordance with predefined policies, security, and SLAs. Also called "services," services are logically distinct from "core services". Services focus on business or mission objectives.
Service consumer	The application or system consuming a service. Also called *consumer*.
Service deregistration	The act of deleting an entry from the SWIM NSRR.
Service discovery	The act of discovering a service. Also referred to as "discovery".
Service-oriented	Pertaining to a service-oriented architecture.
Service-oriented architecture (SOA)	An approach to integrate applications running on heterogeneous platforms using industry-wide acceptable standards. Each application is exposed as one or more services where each service provides a particular functionality. Services (applications) communicate with each other in a coordinated sequence that is defined by a business process.

Service provider	The application or system providing a service. Also called *producer or provider*.
Service registration	The act of creating an entry in the SWIM Service Registry.
Service Registry	See NAS Service Registry Repository
Subscriber	A consumer of a publication service.
Subscription	The process of becoming a subscriber to a publication service. Subscription consists of subscription administration and subscription activation.
Subscription activation	The act of initiating dissemination of publication data and notifications to a subscriber. Subscription activation follows subscription administration, and can occur during either design-time or runtime.
Subscription administration	The act of administering a subscription, including authorization, access list and other database updates, etc. Subscription administration occurs during design-time.
SWIM	System-Wide Information Management. A service-oriented environment for implementing and operating software-based systems that enables information sharing.
SWIM application	A software-based system authorized to provide and/or consume services in the SWIM environment. Also referred to as “application.”
SWIM application developer	A person or organization responsible for developing and maintaining an application. Also referred to as “application developer.”
SWIM business service	Service made available (or exposed) by an end system that can be accessed by other end systems via the SWIM ESB. Also referred to as “business service”
SWIM-enabled service	A service that may be accessed via SWIM.
SWIM core services	The fundamental SWIM mechanisms that enable information sharing: Interface Management, Messaging, Enterprise Service Management (ESM), and Security. These services are solution-agnostic (not limited to a single process or solution environment) and have a high degree of autonomy so that they support reuse. Also referred to as “core services.”
SWIM core services infrastructure	Hardware and software elements that provide the SWIM core services. Also referred to as “core services infrastructure.”
SWIM core services provider	The organization responsible for SWIM core services functionality and infrastructure. Segment 2a services are obtained as value-added services from the FTI provider.
SWIM ESB	Net-centric messaging capability provided via the NEMS nodes
SWIM Service Registry	See NAS Service Registry Repository

User Program	Denotes either a producer or a consumer program; also a program whose application system acts both as a producer and a consumer
Web Service	Allows applications (possibly on a variety of computers using different operating systems) to share business logic, data and processes through a programmatic interface across a network.
Web Service Description Document (WSDD)	A web service is deployed into a message-processing node using an XML-based deployment descriptor file known as a WSDD. WSDD describes how the various components installed in the node are to be chained together to process incoming and outgoing messages to the service.
Web Service Description Language (WSDL)	WSDL is an XML-based interface description language that is used for describing the functionality offered by a web service. A WSDL description of a web service provides a machine-readable description of how the service can be called, what parameters it expects, and what data structures it returns
XML	XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable
XSLT	XSLT is a language used to convert XML documents into other XML documents and other objects

D.2 List of Abbreviations

AIXM	Aeronautical Information Exchange Model
ANSP	Air Navigation Service Provider
AOC	Airline Operations Center
API	Application Program Interface
ARTCC	Air Route Traffic Control Center
ASDE-X	Airport Surface Detection Equipment-Model X
ATC	Air Traffic Control
ATM	Air Traffic Management
AWG	Architecture Working Group (SWIM)
BPEL	Business Process Execution Language
CBR	Content Based Routing
CINP	Communications, Information and Network Programs
CLIN	Contract Line Item Number
CMS	Common Message Set
COI	Community of Interest

ConOps	Concept of Operations
COS	Consumer On-Ramping Service
COTS	Commercial Off The Shelf
COTSWG	COTS Working Group (SWIM)
CY	Calendar Year
DEX	Data Exchange (NEMS equipment)
DHS	Department of Homeland Security
DN	Distinguished Name
DNS	Domain Name System
DoD	Department of Defense
DTS	Dedicated Telecommunications Services
EA	Enterprise Architecture
ES	Exchange Service
ESB	Enterprise Service Bus
ESM	Enterprise Service Management
FAA	Federal Aviation Administration
FIXM	Flight Information Exchange Model
FNTB	FTI National Test Bed
FOC	Flight Operations Centers
FTI	FAA Telecommunications Infrastructure
FTSD	FTI Telecommunications Service Document
FY	Fiscal Year
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
ICD	Interface Control Document
IP	Internet Protocol
IRD	Interface Requirements Document
ISSM	Information Systems Security Manager
IT	Information Technology
JCA	Java Connector Architecture
JMS	Java Messaging Service
JMS-C	JMS Subscription Consumer
JMS-P	JMS Publishing

KSN	Knowledge Services Network
LAN	Local Area Network
MTOM	Message Transmission Optimization Mechanism
NAS	National Airspace System
NBPS	NAS Boundary Protection Service
NEMC	Network Enterprise Management Center
NEMS	NAS Enterprise Messaging Service
NESG	NAS Enterprise Security Gateway
NextGen	Next Generation Aviation System
NOCC	Network Operations Control Center
NPE	Non Person Entity
NSRR	NAS Service Registry/Repository
NTP	Network Time Protocol
OCAS	Oceanic Conflict Advisory System
OCC	Operations Control Center
OEM	Oracle Enterprise Manager
OER	Oracle Enterprise Repository
OES	Oracle Entitlement Server
OGC	Open Geospatial Consortium
OID	Oracle Internet Directory
Ops	Operations
OSB	Oracle Service Bus
PIV	Personal Identity Verification
PNOCC	Primary Network Operations and Control Center
POS	Producer On-Ramping Service
POX	Plain Old XML
PTP	Precision Time Protocol
QoS	Quality of Service
R&D	Research and Development (domain)
REST	Representational State Transfer
RM	Reliable Messaging
RMA	Reliability, Maintainability, Availability
SAML	Security Assertion Markup Language

SAP	Service Access Point (for NEMS)
SDP	Service Definition Point (for FTI)
SIP	SWIM Implementing Program
SLA	Service Level Agreement
SLM	Service Lifecycle Management
SLMP	Service Lifecycle Management Processes
SLO	Service Level Objective
SLOC	Software Lines of Code
SME	Subject Matter Expert
SMF	SWIM Management Function
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SOC	Security Operations Center
SOCC	Security Operations Control Center
SSC	System Support Center
STDDS	SWIM Terminal Data Distribution System
SWIM	System Wide Information Management
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TPP	Telecommunication Program Plan
TRACON	Terminal Radar Approach Control
U.S.	United States
UDDI	Universal Description Discovery and Integration
URL	Uniform Resource Locator
USI	Unique Service Identifier
VNTSC	Volpe National Transportation System Center
VPN	Virtual Private Network
WAN	Wide Area Network
WJHTC	William J Hughes Technical Center
WS	Web Service
WS-C	Web Service Consumer
WSDD	Web Service Description Document
WSDL	Web Service Description Language

WS-P	Web Service Producer
WSRD	Web Service Requirements Document
WS-RM	Web Services Reliable Messaging
WXXM	Weather Information Exchange Model
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformations